

Aalto University  
School of Electrical Engineering  
Degree Programme of Bioinformation Technology

Arttu Modig

# A Metric for Human Motor Capacity

Master's Thesis  
Helsinki, 20. toukokuuta 2016

Supervisor:	Prof. Jouko Lampinen
Advisor:	Ph.D. Teemu Roos

<b>Author:</b>	Arttu Modig		
<b>Title:</b>	A Metric for Human Motor Capacity		
<b>Date:</b>	20. toukokuuta 2016	<b>Pages:</b>	viii + 78
<b>Major:</b>	Computational and Cognitive Bio-science	<b>Code:</b>	IL3003
<b>Supervisor:</b>	Prof. Jouko Lampinen		
<b>Advisor:</b>	Ph.D. Teemu Roos		
<p>In this thesis, a novel metric for measuring human motor performance is presented. The intuition is derived from Fitts' law, but unlike Fitts' law, the metric is generalizable to continuous, free-form, full-body motion, which is reproducible. The applications of interest lie in human-computer interaction (HCI), kinesiology, sports science, and user authentication.</p> <p>For a background, the thesis presents the Fitts' law and its use as an evaluation tool in HCI. The extensions and restrictions are briefly presented. As the human motion is captured through sensor devices of different techniques, the choice of the sensor is important as it affects the data available for the metric's evaluation. The motion data acquisition is described with different sensor systems that have been used or tried out in the experiments.</p> <p>As the sensor space usually contains redundant motion information compared to the user's inherent motion space, the information needs to be reduced through unsupervised machine learning techniques. As for preprocessing, Principal Component Analysis (PCA), Probabilistic PCA and Gaussian Processes Latent Variable Model (GP-LVM) were used for dimensionality reduction, and Canonical Time Warping (CTW) was used for temporal alignment. Throughput was then calculated using mutual information.</p> <p>Experimental evaluation and assessment for the metric was made. Classical ballet was used as reference data, with throughputs ranging from 213 to 590 bps using GP-LVM. Comparisons to Fitts' law were made with a cyclical tapping task. Bimanual in-air gesturing was used to examine some well-known motor-perceptual phenomena, and the metric showed responsiveness to laterality and perceptual distraction. Also, several diagnostics were made and the problems of the framework were assessed.</p>			
<b>Keywords:</b>	information theory, human-computer interaction (HCI), Fitts' law, machine learning, dimension reduction, throughput		
<b>Language:</b>	English		

Aalto-yliopisto  
Sähkötekniikan korkeakoulu  
Bioinformaatioteknologian koulutusohjelma

DIPLOMITYÖN  
TIIVISTELMÄ

<b>Tekijä:</b>	Arttu Modig		
<b>Työn nimi:</b>	Ihmisen motorisen suorituskyvyn mittaaminen		
<b>Päiväys:</b>	20. toukokuuta 2016	<b>Sivumäärä:</b>	viii + 78
<b>Pääaine:</b>	Laskennallinen ja kognitiivinen biotiede	<b>Koodi:</b>	IL3003
<b>Valvoja:</b>	Prof. Jouko Lampinen		
<b>Ohjaaja:</b>	FT Teemu Roos		
<p>Tämä opinnäytetyö esittelee uuden metriikan ihmisen motorisen suorituskyvyn mittaamiseksi. Intuitio on peräisin Fittsin laista, mutta toisin kuin Fittsin laki, kehitetty metriikka on yleistettävissä jatkuvaan, vapaamuotoiseen ja koko kehoa koskevaan liikkeeseen, joka on toistettavissa. Metriikan sovellukset ovat HCI-tieteessä, kinesologiassa, liikuntatieteessä ja käyttäjien autentikaatiossa.</p> <p>Työn taustatietoina esitetään Fittsin lain teoria ja sen käyttö HCI-tieteessä. Fittsin lain laajennukset ja rajoitukset esitetään lyhyesti. Koska ihmisliikettä voidaan tallentaa erilaisilla sensoreilla ja tekniikoilla, on tallennusmetodin valinta tärkeää, koska se vaikuttaa kerätyn tiedon muotoon. Liiketiedon kerääminen kuvataan eri laitteistoilla, joita on käytetty tässä tutkimustyössä.</p> <p>Koska sensoritieto on yleensä redundanttia verrattuna käyttäjän liiketietoon, täytyy sensoritieto pelkistää ohjaamattomilla koneoppimistekniikoilla. Tiedon esiprosessointina käytettiin pääkomponenttianalyysiä (PCA), probabilistista PCA:ta ja gaussisten prosessien latenttia muuttujamallia (GP-LVM) pelkistämään sensoritietoa, ja kanonista aikaväristystä (CTW) käytettiin sekvenssien ajalliseen linjaukseen. Suoritusteho laskettiin keskinäisellä informaatiolla.</p> <p>Metriikan toimivuus arvioitiin kokeellisesti. Referenssiaineistona käytettiin klassista balettia, jonka suoritustehoksi laskettiin 213–590 bittiä per sekunti GP-LVM-menetelmällä. Vertailut Fittsin lakiin tehtiin syklistä naputuskokeella. Kaksikätistä elehtimiskoetta käytettiin tutkimaan joitakin tunnettuja motorisia hahmottamisilmiöitä, ja metriikka osoitti herkkyyttä lateralisuudelle ja hahmotushäiriöille. Työssä diagnosoitiin myös metriikan soveltamista ja ongelmia arvioitiin.</p>			
<b>Asiasanat:</b>	informaatioteoria, HCI, Fittsin laki, koneoppiminen, dimensio-reduktio, suoritusteho		
<b>Kieli:</b>	Englanti		

# Acknowledgements

This Master's Thesis is the culmination of the long work which took place in HIIT Helsinki Institute of Information Technology and Max-Planck-Institut für Informatik in Saarbrücken, Germany. First and foremost, I am very grateful for the mentorship and innumerable advices from Ph.D. Antti Oulasvirta and Ph.D. Teemu Roos. My visit to Saarbrücken hosted by Antti in summer 2012 was very inspirational and an eye-opener. The guidance of Teemu was excellent, and it has been a delight to work in his research group, as with Antti's.

Very warm thanks for Prof. Frank Pollick for hosting me in Glasgow in Autumn 2011, and helping me to work through the Vicon motion capture system. I wish to thank Naree Kim for the beautiful (and very excellent) ballet.

I am also grateful for Prof. Jouko Lampinen for helping me to redefine and improve this thesis, and for D.Sc. Antti Honkela for helping me in Gaussian Process modeling. I wish to thank for many other colleagues in HIIT who have helped me during this work.

This thesis would not be ready without the support of my dear friends and my loving family. Special thanks for the advice from M.Sc. Tommi Vatanen, M.Sc. Sakari Cajanus, and the rest of #terassillekohta. Time seems to fly by so fast with you, it feels crazy.

Lämmin kiitos rakkaille vanhemmilleni mittaamattomasta tuesta, kärsivällisyydestä ja ymmärryksestä.

Espoo, 20. toukokuuta 2016

Arttu Modig



# Glossary

## Abbreviations and Acronyms

ARD	Automatic Relevance Detection; automatic inference of dimensions in Bayesian learning
bps	Bits Per Second
CCA	Canonical Correlation Analysis; a method for finding maximum correlation between two data sets
CTW	Canonical Time Warping; a time series alignment algorithm
DTW	Dynamical Time Warping; a time series alignment algorithm
FPS	Frames per Second
GP-LVM	Gaussian Processes Latent Variable Model; a dimension reduction and data visualization method
GUI	Graphical User Interface
HCI	Human-Computer Interaction; the multidisciplinary study of interaction between humans and computers
ID	Index of Difficulty; the difficulty of an aiming task
IP	Index of Performance; the rate of human motor performance
IR	Infra-Red light
MOCAP	Motion Capture
MT	Movement Time; the time to perform a single aiming task
TP	Throughput; the rate of human motor performance
PCA	Principal Component Analysis; a dimension reduction and data visualization method
PHP	PHP: Hypertext Preprocessor; a server-side general-purpose programming language
PPCA	Probabilistic Principal Component Analysis

## Symbols and Operators

$\exp(\cdot)$	Exponent function
$\mathbb{E}[\cdot]$	Expected value
$C$	Channel capacity
$H(\cdot)$	(Discrete) entropy
$h(\cdot)$	Differential entropy
$I(\cdot; \cdot)$	(Mutual) information
$I_d$	Index of difficulty
$J(\cdot)$	Cost function to be minimized
$D(q  p)$	Relative entropy or Kullback-Leibler divergence of $q$ from $p$
$k(\cdot, \cdot), \mathbf{K}$	Kernel or covariance function
$\mathcal{L}(\cdot)$	Likelihood function
$\log_2$	Logarithm of base 2
$m(\cdot)$	Mean
$N$	Signal noise
$n, N$	The index of data vector and the number of samples
$\mathcal{N}(\mu, \Sigma)$	A normal distribution with mean $\mu$ and (co)variance $\Sigma$
$O(\cdot)$	Big O asymptotic notation; upper bound of the growth rate of the function
$P$	Signal power
$p(x)$	Probability density function of $x$
$p(x y)$	Probability density function of $x$ given $y$
$R$	Radius
$\mathcal{R}^{x \times y}$	A real-valued matrix with $d$ rows and $y$ columns
$\text{tr}(\mathbf{X})$	Trace of matrix $\mathbf{X}$
$W_e$	Effective width
$x$	Scalar value
$\hat{x}$	Predicted or estimated value of $x$
$\mathbf{x}$	Vector with elements $x_i$
$\mathbf{x}^T$	Transpose of $x$
$  \mathbf{x}  $	$\ell^2$ norm of vector $\mathbf{x}$
$\mathbf{X}$	Matrix with elements $x_{ij}$
$\theta$	Model parameter(s)

# Contents

<b>Glossary</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Fitts' Law</b>	<b>4</b>
2.1 Fitts's Paradigm . . . . .	4
2.2 Fitts' Law as an Evaluation Tool . . . . .	6
2.3 Shannon Formulation . . . . .	8
2.4 Extensions and Restrictions of Fitts' law . . . . .	9
<b>3 Motion Data Acquisition</b>	<b>11</b>
3.1 Motion Capture . . . . .	11
3.2 Depth Cameras . . . . .	14
3.3 Mouse and Touchscreen . . . . .	16
<b>4 Information Estimation</b>	<b>17</b>
4.1 Motion Alignment . . . . .	17
4.1.1 Canonical Correlation Analysis . . . . .	17
4.1.2 Dynamic Time Warping . . . . .	18
4.1.3 Canonical Time Warping . . . . .	18
4.2 Dimension Reduction by Machine Learning . . . . .	20
4.2.1 Principal Component Analysis . . . . .	20
4.2.2 Probabilistic PCA . . . . .	21
4.2.3 Gaussian Process Latent Variable Model . . . . .	23
4.2.4 Learning in the GP-LVM . . . . .	25
4.3 Information Estimation . . . . .	27
4.3.1 Complexity Estimation . . . . .	27
4.3.2 Entropy . . . . .	28
4.3.3 Relative Entropy . . . . .	30
4.3.4 Mutual Information . . . . .	30
4.3.5 Mutual Information of a Bivariate Normal Distribution	33

4.3.6	Throughput . . . . .	34
<b>5</b>	<b>Implementation</b>	<b>35</b>
5.1	Capturing of Motions and Gestures . . . . .	35
5.1.1	Vicon and PhaseSpace . . . . .	35
5.1.2	Mouse Traces . . . . .	36
5.1.3	Microsoft Kinect . . . . .	36
5.2	Preprocessing of the Data . . . . .	36
5.3	Algorithms . . . . .	37
5.4	Server Implementation . . . . .	38
<b>6</b>	<b>Results and Evaluation</b>	<b>40</b>
6.1	Classical Ballet . . . . .	40
6.2	Cyclical Tapping With a Mouse . . . . .	43
6.3	Bimanual In-Air Gesturing . . . . .	46
6.4	Metric's Assessment . . . . .	50
6.4.1	Complexity Model Diagnostics . . . . .	50
6.4.2	Frame Rate Manipulation . . . . .	52
6.4.3	Effect of Noise . . . . .	53
6.4.4	Static and Repetitive Motion . . . . .	55
<b>7</b>	<b>Discussion</b>	<b>57</b>
7.1	Applications . . . . .	58
7.2	Future Work . . . . .	59
<b>8</b>	<b>Conclusions</b>	<b>61</b>
<b>9</b>	<b>Appendix: Gaussian Channel Capacity</b>	<b>75</b>
9.1	Channel Capacity . . . . .	75
9.2	Gaussian Channel . . . . .	76

# Chapter 1

## Introduction

Measuring human's motor capabilities has always been a part of human culture. Since the dawn of sports, people have been competing with each other in skills that require a high level of motor-cognitive competence. Besides sports and other physical games, many artistic and musical skills, such as playing an instrument or performing a dance, require a high level of motorical capabilities. A sufficiently high level of motorical skill is usually deemed to be needed to be able to compete or play in the highest class of dancing, playing, and sports.

Evaluation of such skills is a whole discipline. In sports, where the scored performance is complex, there is a judge or a whole jury to evaluate the level of skill. In sophisticated competitions, there needs to be experts, usually expert performers themselves, that have the required knowledge to be able to evaluate others. Thus, there are virtually no objective "hard" metrics that could help with the evaluation process or even standardize it. The scoring process is commonly a jury of experts, relying more often to qualitative and subjective than quantitative and objective information. Measuring complex performance relies usually on keen human surveying that can be prone to errors and changes in perception.

In contrast, the modern world has changed into a data globe of vast quantitative information. Rate of communication and the exchange and extraction of information has never been faster and more efficient, which has also resulted in the explosion of new knowledge in various fields. The 21st century is perhaps best characterized by the emerging *big data* in many fields, such as genomics, physics and Internet search. Processing the large and complex data sets is difficult, but the cost of such efforts is getting smaller, and valuable analysis is more and more feasible. Extracting knowledge from complex high-dimensional data sets is one of the key areas in modern statistics and machine learning.

Nonetheless, the entire so-called information age relies on the standard blocks of information: the bits. 1's and 0's have mathematically formalized many scientific concepts and the flow of information. Yet, it requires hard work to transform qualitative knowledge into quantitative, and to create general metrics that can translate the subjective knowledge into objective information.

This transitional problem is perhaps most clear in psychological and cognitive sciences. For example, most medical data is extractable through chemical or physical sensors, and as such, it is already feasible for evaluation, but creating strict objective tests to evaluate and standardize human performance is a difficult task. In many behavioral tasks, the problem is indeed very complex [82, 106]. Reflecting the triumph of computers and robots, many question also the human performance in similar metrics. What is the information processing in the human brain? What is the maximum rate of human communication? What is the information capacity of the human motor system?

In the wake of Claude E. Shannon's ground-breaking work on information theory in 1940s, there was a huge scientific interest to apply the theory in psychology [6], which Shannon regarded as a bandwagon effect [86]. Nonetheless, Paul Fitts made a fundamental observation in 1954 [27] that the rapid movement performance of an arm can be modeled in a prescribed task, and this mathematical model can also evaluate the information capacity of aimed movements. Even nowadays, *Fitts' law* and its extensions remain as some of the few "hard" predictive models in human-computer-interaction, but the applications are very restricted.

In this thesis, I present and extend a novel research work [74], which attempts to create a general metric for measuring the performance rate, *throughput*, of human motor control. The initial paradigm of Paul Fitts remains; since human responses are continuous variables, the information capacity is only limited by the statistical variability, or noise, of the responses. With the motion capture techniques of today, we have the tools to precisely track the continuous movements of a full human body. Thus, we have the data that enables us to estimate the information content of full-body motion available to an external viewer.

The applications of a general throughput metric for human motor expressiveness and control are tentatively vast. Besides as a general tool for human factors and human-computer interaction, we hypothesize that the metric could be also useful in sport science and medical science. For example, comparing to a movement reference given by a skilful *t'ai chi* teacher, the metric would reveal the exact shortcomings in each student's repetitions. Or a clinical study, where the deterioration or improvement of patients with

neuromotor impairments could be tracked exactly with a few motorical tests. The human perception has its limitations, but the precision in information world is in practise limitless. Even small deteriorations or variations in motor performance—that would otherwise escape human senses—can be revealed with high-performance modeling and information theory.

Despite the almost intuitive idea, creating a robust framework for the throughput metric is a hard problem to tackle. First, it requires sophisticated hardware and software to extract the motion data. Second, the motion data has to be standardized, which requires robust *machine learning* algorithms. Third, the *mutual information* has to be estimated. With state-of-the-art methods, the whole process can be very complex and slow. For fast analysis and evaluation, we have to use rough approximations that will overestimate the motion’s inherent information content. But even with grossly overestimating the throughput values, the metric can be used to cross-compare and evaluate motions within similar style and data format.

The structure of this thesis is the following: In Chapter 2, as a background, I introduce the paradigm of Fitts’ law. In Chapter 3, I describe the different methods of acquiring motion data of various kinds. In Chapter 4, I describe the different mathematical methods that are required in the estimation of motion’s information. After these theoretical chapters, the implementations will be presented in Chapter 5. Then, the evaluation methods and results are presented in Chapter 6. Finally, the discussion about the current framework, future work and prospects are in Chapter 7, along with the conclusions in Chapter 8.

## Chapter 2

# Fitts' Law

The human motor control is the process of using the neuromuscular system to activate and coordinate the muscles and limbs under the performance of certain motor skills. It is based on the integration of sensory information, both about the environment and the body, to determine the appropriate neuromuscular actions to generate some desired movement. As the process requires cooperation between the central nervous system and the musculoskeletal system, it is as such a problem of information processing, coordination, mechanics, physics, and cognition. Successful motor control is crucial to interacting with the environment, and a high performance of a person's general motor control and skills is advantageous.

Since Claude E. Shannon's information theory in the late 1940s, the theory was quickly applied in the fields of cognitive and behavioral sciences, in relation to sensory, perceptual, and perceptual-motor-functions. The work by Paul Fitts in 1950s extended the framework to human motor control [27, 28], and it has become the dominant paradigm in studying the information capacity of the human motor system. The primary application of Fitts's paradigm, commonly known as *Fitts' law*, is the analysis and evaluation of user interfaces in human-computer interaction (HCI) [56, 95, 109, 110]. For example, the paradigm was one main motivation in the development and adoption of the computer mouse over the joystick [5, 14].

## 2.1 Fitts's Paradigm

In the pioneering work, Fitts was interested in *discrete aimed continuous movements*, that is, movement where a pointer (arm, finger, display cursor, etc.) is moved onto a spatially expanded target. A common example is moving a mouse cursor on top of a button and clicking it on a computer display.



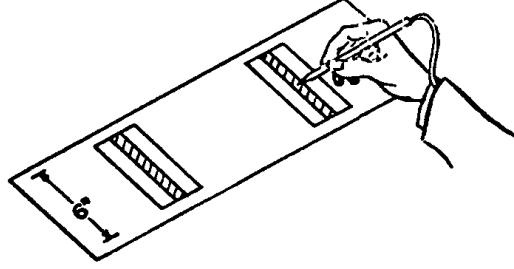


Figure 2.1: Reciprocal tapping apparatus after Fitts [27]. The task was to hit the center plate in each group alternately without touching either side (error) plate.

Originally, one of Fitts' lab experiments [27] had a setup of a physical apparatus with metal plates and metal-tipped stylus pens, represented in Figure 2.1. The instructions for the tapping were to hit the plates alternatively in a time limit, with emphasis on accuracy rather than speed. This long reciprocal tapping effectively minimized the hindering effect of motor-cognitive reaction time.

Fitts's specific hypothesis was: *the average time per response will be directly proportional to the minimum average amount of information per response demanded by the particular conditions of amplitude and tolerance*. Formulated as a linear relationship, the time per response is commonly dubbed as the *movement time* (MT), and the amount of information per response, the number of bits to fulfill the task, is dubbed the *index of difficulty* (ID):

$$MT = b \cdot ID. \quad (2.1)$$

where  $b$  is the unknown parameter. Fitts' original definition of index of difficulty  $I_d$  was

$$I_d = -\log_2 \left( \frac{W_S}{2A} \right) \text{ bits/response}, \quad (2.2)$$

where  $W_S$  is the target width and  $A$  is the distance to the target or "amplitude" – a rather confusing term which was used by Fitts. Equation (2.2) is now commonly known as

$$ID = \log_2 \frac{2D}{W} \text{ bits}, \quad (2.3)$$

where  $D$  is the distance to the target center, and  $W$  is the width of the target. When  $R = \frac{1}{2}W$ , the ID can also be presented as

$$ID = \log_2 \left( \frac{D}{R} \right) = \log_2 \left( \frac{D}{\Delta D} \right) \text{ bits}, \quad (2.4)$$

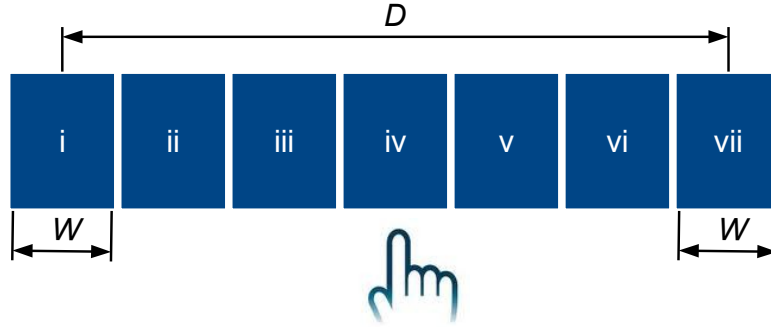


Figure 2.2: The Fittsian aiming task. Under the Fitts's model, it is possible to hit each bin with the same probability when sending a message. The motor capacity is determined, when the subject cannot hit targets farther than  $i$  and  $vii$  in the same time interval as  $i$  and  $vii$  with enough accuracy.

where  $D$  is the length of the required movement, and  $\Delta D$  is the error or inaccuracy of the movement.

ID Equation (2.4) in essence the information needed to complete an aiming task. According to the properties of *self-information* (discussed in Section 4.3.4), the logarithm of the number of bins (which are distinct possible “messages”) of uniform probability gives us the information needed when one of the bins is “selected”. Under the Fitts's paradigm, the subject has at least the same efficiency of hitting all the bins as hitting the most difficult target bin. The paradigm is shown in Figure 2.2 where the target bin width is  $W = 2R = 2\Delta D$ .

## 2.2 Fitts' Law as an Evaluation Tool

In the time of Fitts's original experiment, the true dispersion of the target hits and misses was not available, as Fitts only knew the error-rate in each procedural task. With the computer mouses, displays and touchscreens of nowadays, the dispersion of the target taps can be modelled more accurately than before. With this information, it is possible to calculate the effective width  $W_e$  of the target, which is based on a certain factor of the standard deviation. Usually in the literature [57, 101, 110], the factor is  $W_e = \sqrt{2\pi e}\sigma \approx 4.133\sigma$ , which is 96.12 % of the weight of the normal probability distribution.

Using this effective width  $W_e$  it is also possible to calculate the effective (or adjusted) index of difficulty  $ID_e$ . As effective width depends on the variance of the hits, which have the property to be Gaussian [108], Fitts' law

also relates that the speed-accuracy trade-off of motor-cognitive control is independent of the throughput as it is the constant motor control capacity in aiming tasks of the same type. This has been proved empirically by Fitts and Radford [29] and MacKenzie and Isokoski [57].

Fitts's experimental setup had no reaction times involved, but in an experiment where reaction times are involved, a bias time term  $a$  has to be added. Thus, the model has the popular form and is known as the *Fitts' law*:

$$MT = a + b \cdot ID. \quad (2.5)$$

The term  $a$  can also take into account factors, that are not modelled by the Fittsian aiming task itself.

Fitts coined the term *index of performance* (IP) as the performance rate in bits per second:

$$IP = \frac{ID}{MT}. \quad (2.6)$$

which used the original model (2.1) which does not have the reaction time term  $a$ , thus defining IP as

$$IP = \frac{1}{MT} \log_2 \frac{2D}{W} \text{ bits/s}, \quad (2.7)$$

where MT is the movement time per response,  $W$  is the width of target and  $D$  the distance to the target. IP is in essence the rate of performance ability, also called as the *throughput*. From the linear regression problem (2.5), the throughput in a Fittsian movement task is usually defined as

$$TP = \frac{1}{b} \text{ bits/s}, \quad (2.8)$$

though an International Standard *ISO 9241-9* was created which defines throughput also as

$$TP = \frac{ID}{MT}. \quad (2.9)$$

The distinction between these lies in the premiss whether the parameter  $a$  in seconds—often denoted as the reaction time in the experimental task—is part of the Fittsian movement time, thus whether  $a$  affects the throughput that is estimated from a regression problem.

When Fitts' law Equation (2.5) is used in a linear regression problem, the throughput  $1/b$  can be approximated in an experiment with a least-squares method, where the relationship between the difficulty of an aiming task (ID) and the time to complete a particular response (MT) can be observed. In the literature this has been a standard method of evaluating a certain input

device based on continuous aiming [14, 56, 95]. For example, selecting targets with a mouse yields a range of throughput of 3.7–4.9 bps while selecting targets with a touchpad yields a range of 1.0–2.9 bps [95].

## 2.3 Shannon Formulation

A notion of worth has to be said about the possible premiss which Fitts had, that the decision to not move at all does not convey any information, that is, selecting the bin where the hand is already placed does not require any additional information (see Figure 2.2). In contrast to the Shannon's *channel capacity* theorem, all distinct possible messages count. Therefore, as an analogue to Shannon's distinct codewords, the number of possible bins (messages) is  $(D + W)/W$ , and the index of difficulty becomes

$$\text{ID}_{\text{Shannon}} = \log_2 \left( \frac{D}{W} + 1 \right), \quad (2.10)$$

and the throughput (without the time bias  $a$ ) becomes

$$\text{TP}_{\text{Shannon}} = \frac{1}{\text{MT}} \log_2 \left( \frac{D}{W} + 1 \right). \quad (2.11)$$

MacKenzie [55] was first to adopt this formulation, though according to MacKenzie, it was based on a “direct analogy” of Shannon's Theorem 17 (Shannon–Hartley theorem) [85], the channel capacity of a communications channel:

$$C = W \log_2 \left( \frac{P}{N} + 1 \right), \quad (2.12)$$

where  $W$  is the bandwidth of the channel in hertz,  $P$  is the transmitter power, and  $N$  is the noise power. Compare this to Shannon-TP Equation (2.11). MacKenzie dubbed his version of Fitts' law as the Shannon model:

$$\text{MT}_{\text{Shannon}} = a + b \log_2 \left( \frac{D}{W} + 1 \right). \quad (2.13)$$

Though Fitts mentioned Shannon's Theorem 17 in the original paper [27], he was for some reason reluctant to adapt it directly, perhaps for the conceptually difficult implication that a “no-movement” could be also interpreted as a message.

This *Shannon formulation* has been a common alternative formulation of Fitts' law in the HCI literature especially after MacKenzie et al. [58].

Although the similarity of Equations (2.11) and (2.12) is prominent, the real justification of Equation (2.13) has been criticised by Drewes [24] and Hoffmann [39]. Nevertheless, MacKenzie [59] defended the empirical validity of the Shannon version.

In this thesis, the Shannon's sphere packing analogy to channel capacity is presented in Appendix 9.2, and it can be argued that the Shannon-ID (2.10) is also justified, as it reflects the general difficulty of the aimed pointing task—the motor-cognitive output information needed to solve the task—which is logarithmically proportional to the number of distinct possible messages the subject can send. This was also Fitts's original idea as the information theoretical basis [27].

Initiative work on the exploration of differentiable motion space has been done by Williamson and Murray-Smith [105], though in which the focus was on identifying motion *variability* in the full joint user-sensor motion space, here the focus is on motion *reliability* and *complexity* in similar movement tasks.

## 2.4 Extensions and Restrictions of Fitts' law

Fitts' law has been extended into continuous aimed movements, which first covered just the path width and length [1, 2], but were later extended also to curvature [53]. However, these extensions to Fitts' law have no interpretation in information theory. Some research has also considered the information capacity of bimanual control [61].

The general restriction of Fitts' law paradigm and its extensions is that the experimental conditions are *prescribed* to a high degree by the experimenter, and limited to just few body parts. Also, the information capacity can be evaluated only from aimed movements where just the end points matter. Thus, the information being measured is tantamount to the subject's ability to motorically conform to the extrinsic constraints, excluding entirely free movement that is irrespective of its absolute position and where the goal is to produce a certain shape or pattern. Many skilled activities also utilize multiple body parts or the full body, such as dance and sports. Most also involve compound tasks where multiple types of movement are performed simultaneously and sequentially.

Due to these limitations, it is easy to see that the Fitts' law paradigm is not suitable for the study of skilled motor action; the action that can be expected to contain and communicate the most information. Therefore, the main motivation of this thesis work is to create a new paradigm based in information theory that would take into account motion that has multiple

and continuous movement points, and where the emphasis is on shapes and patterns created by the subject herself, with as few experimental restrictions as possible.

## Chapter 3

# Motion Data Acquisition

Motion capture is the process of recording the movement of objects or people. Specifically, the analysis of human actions by a computer is gaining more and more interest and it has many applications in different areas. The computer vision-based human motion capture can be divided into three major application areas: surveillance, control, and analysis. Classical *surveillance applications* are people counting or crowd flow analysis, or analysis of abnormal activity and behaviour in airports or other high-security areas. In *control applications* the motion or pose parameters are used to control something. The most well-known control applications among the general public are perhaps gaming gear, for example Microsoft's Xbox Kinect [65], Sony's PlayStation Eye [94], and the novel virtual reality device Oculus Rift [71]. More generally, control applications are a topic of human-computer interfaces. Also in the entertainment and film industry motion capture is used to generate and define personalized computer graphic models to make the productions more believable. Third major application area of human motion capture are *analysis applications*, such as automatic diagnostics of patients with orthopedic or neuromotoral problems, or analysis and optimizations of an athlete's performances. More generally, applications in the study of kinesiology and human factors. [67, 68]

### 3.1 Motion Capture

According to a general survey [67], the functional taxonomy of motion capture research and also the general structure of the systems can be separated as follows: initialization, tracking, pose estimation, and recognition. These are general aspects which reflect the various types of information being processed. Before a system is ready to process data it needs to be *initialized*

which may include calibration or other model parameter selection. Next, the motion of the subject(s) is *tracked* which implies a way of segmenting the subject from the background and finding the correspondies between segments in consecutive frames. The *pose* of the subject's body often needs to be estimated, as the skeletal model is important in many graphical representations and may be procesed further. Some higher level knowledge of the human body is typically used in *pose estimation*. The final step in the process may be analyzing the pose or other parameters in order to *recognize* the indentity of the subject, or the actions and behaviors performed by the subject.

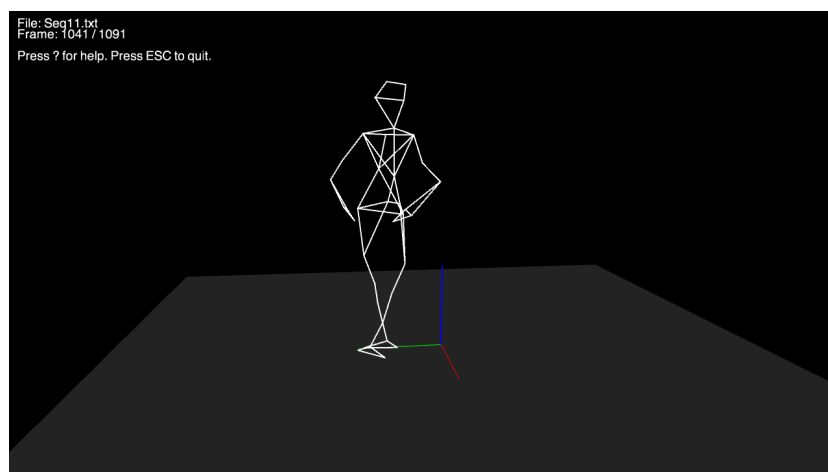
The majority of vision-based tracking systems assume *a priori* a humanoid kinematic structure comprising a fixed number of joints with specified degrees-of-freedom. The kinematic initialization is then limited to the estimation of limb lengths. The known correpondence between markers and limbs together with reconstructed 3D marker trajectories during movement are then used to accurately estimate limb lengths. Sometimes hard constraints on left-right skeletal symmetry are imposed during estimation. Anthropometric constraints between ratios of limb lengths are imposed to allow estimation of the kinematic structure up to an unknown scale factor. It is also possible to make a direct estimation of the kinematic structure from sequences of a moving person [47, 48, 93]. Increasingly, human motion capture sequences from high-accuracy commercial marker-based systems have been used to learn a prior model of human kinematics and specific motions to provide constraints for subsequent tracking. Similarly, motion capture data bases [17, 19] have been used to synthesize image sequences with known 3D pose correspondence to learn *a priori* the mapping from image to pose space for reconstruction. [68]

Many commercial marker-based systems provide high-resolution tracking and pose estimation, whereas markerless systems are generally less accurate. The markers yield a high contrast in the images and provide a robust representation [91]. If more than one camera is used, a 3D representation can be obtained via triangulation [69]. In recent years, the marker-free systems have developped considerably, but still the current approaches are limited to gross-body pose estimation of torso, arms and legs and do not capture detailed movement, such as hand-orientation or axial arm rotation. Multiple hypothesis sampling achieves robust tracking but does not provide a single temporally consistent motion estimate, resulting in jitter which must be smoothed to obtain visually and analytically acceptable results. There still remains a substantial gulf between the accuracy of commercial marker-based and markerless video or depth data based human motion reconstruction. [68]





(a) An example of a motion capture lab on video.



(b) A 3D visualization of the captured pose.



(c) Vicon MX-F40 infrared camera.

Figure 3.1: Vicon motion capture system; an example of active marker-based motion capture.



Figure 3.2: Kinect controller for Xbox 360 video game console.

## 3.2 Depth Cameras

In recent years, the advancements in range (or depth) imaging techniques have considerably affected human body tracking techniques. The task has been greatly simplified by the introduction of real-time depth cameras [33, 35, 45, 78, 90, 113]. Markerless pose estimation is still a tough problem to tackle, and until the launch of Microsoft's Kinect [65] in Figure 3.2, none of the systems ran at interactive rates on consumer hardware while handling a full range of human body shapes and sizes in on-going motion. Depth cameras offer several advantages over traditional light intensity sensors, working in low light levels, giving a calibrated scale estimate, being color and texture invariant, resolving silhouette ambiguities in pose, and greatly reducing the task of background subtraction. Most importantly for the Kinect software, it was straightforward to build a large training dataset cheaply. [89]

The Kinect sensor technology was provided by an Israeli range camera technology developer PrimeSense. The patent [30] describes the measurement of depth as a triangulation process. The laser source emits a single beam which is split into multiple beams by a diffraction grating to create a constant pattern of speckles projected onto the scene. This pattern is captured by an infrared camera and is correlated against a reference pattern. The reference pattern is obtained by capturing a plane at a known distance from the sensor, and is stored in the sensor's memory. The shifts of each speckle compared to the baseline are then measured by a simple image correlation procedure, which yields a disparity image. [44]

The resolution of the Kinect sensor is determined by the resolution of the infrared camera, and it also determines the point spacing of the depth data perpendicular to the camera axis. Since each depth image contains a constant  $640 \times 480$  pixels, the point density will decrease with increasing distance from the sensor. Kinect disparity image contains 1024 levels of disparity, so at a range of 2 meters one level of disparity corresponds to 1 cm depth resolution, while at 5 meters one disparity level corresponds to about 7



Figure 3.3: Leap Motion controller in front of the keyboard and the hand tracking visualization.

cm depth resolution [44]. Taking into account the other imperfections of the sensor, such as errors from occlusions and overexposed surfaces, it is clear that the Kinect sensor provides only very rough data for the motion capture pose estimation, though the system is inexpensive, robust and easy to set up.

In the wake of Kinect, other inexpensive consumer range cameras have been released. Most notably, *Leap Motion* started shipping in July 2013 [52], pictured in the Figure 3.3. The Leap Motion controller is a small USB peripheral device, about the size of a USB flash drive, which is designed to be placed on a physical desktop facing upward. Using two monochromatic infrared cameras and three IR LEDs, the device observes a roughly hemispherical area from about 1 cm up to 1 meter, and it is designed to track hand and finger motions. Due to patent pending, the exact method of capturing the depth image is not known yet. The precision of the controller appears to be very high, but the pose estimation of the proprietary software suffers from occlusions and seems to need a silhouette of the hand to track

the finger joints properly. As the development is still in early stages, the pose estimation may improve in future. As for now, Leap Motion data is too inaccurate to be used in unrestricted hand motion capturing though it offers more degrees of freedom in simple motions than touch surfaces.

### 3.3 Mouse and Touchscreen

The general graphical input mechanisms for computers are vastly dominated by devices restricted to flat 2D surfaces, such as mice and touchscreens. Evaluation of the most effective pointing devices are traditionally based on Fitts' law, for example in a text selection task, a mouse is proved to be more effective than a joystick [14]. It is difficult to determine the true tracking accuracy of a mouse, because there are several other affecting factors than just high DPI and framerate measures. For the purposes of the studies in this thesis, commercial high-end gaming mice are considered to be robust and accurate enough.

In recent years, the development of capacitive touchscreens has been very active. Most major tablet and mobile devices feature a capacitive touchscreen panel, and manufacturers have even developed in-cell touch panels which integrate the production of capacitive sensors in the AMOLED displays [15]. Still the accuracy of different capacitive touchscreen panels has been shown to have a lot of variation, as errors over 1 mm can be typical [73]. One problem when using data from capacitive touchscreens may be that the framerate of detected events is not constant but varies depending on the speed and type of motion.

## Chapter 4

# Information Estimation

### 4.1 Motion Alignment

#### 4.1.1 Canonical Correlation Analysis

Ordinary correlation analysis is dependent on the coordinate system in which the variables are described. There could be a very strong linear relationship between two multidimensional signals which would not be visible if a selected coordinate system would be used. Therefore, the problem is to find the projections of the two signals that give the best cross-correlation.

Canonical correlation analysis (CCA) [41] is a technique of measuring the linear relationship between two multidimensional variables. CCA identifies relationships between two sets of variables by finding the linear combinations of the variables in the first set  $\mathbf{X} \in \mathbb{R}^{n \times d_x}$  that are most correlated with the linear combinations of the variables in the second set  $\mathbf{Y} \in \mathbb{R}^{n \times d_y}$ . This can be viewed as a generalized eigenvector problem where the task is to find a corresponding pair of linear projections (subspaces) that maximize the cross-correlation, whereas Principal Component Analysis (PCA, discussed in Section 4.2.1) operates with a single variate and maximises the variance of projections of the data.

Formally, assuming zero-mean data, CCA finds a combination of the original variables such that the following norm is minimized:

$$\begin{aligned} J_{CCA}(\mathbf{V}_x, \mathbf{V}_y) &= \|\mathbf{V}_x^T \mathbf{X} - \mathbf{V}_y^T \mathbf{Y}\|_F^2, \\ \text{such that } \mathbf{V}_x^T \mathbf{X} \mathbf{X}^T \mathbf{V}_x &= \mathbf{V}_y^T \mathbf{Y} \mathbf{Y}^T \mathbf{V}_y = \mathbf{I}, \end{aligned} \tag{4.1}$$

where  $\mathbf{V}_x \in \mathbb{R}^{b \times d_x}$  is the projection matrix for  $\mathbf{X}$ , similarly for  $\mathbf{Y}$ , and  $\|\cdot\|_F$  designates the Frobenius norm. A pair of canonical variates  $(\mathbf{v}_{x,i}^T \mathbf{X}, \mathbf{v}_{y,i}^T \mathbf{Y})$  is uncorrelated with other canonical variates of lower order, and each successive

canonical variate pair achieves the maximum correlation orthogonal to the preceding pairs. [112]

### 4.1.2 Dynamic Time Warping

Dynamic time warping (DTW) is a technique to find out an optimal alignment between two time series under certain restrictions. Intuitively, the sequences are warped in a nonlinear fashion to match each other. Originally developed for speech recognition [79], DTW has been applied to almost any data that can be turned into sequences.

Formally, given two time series,  $\mathbf{X} \in \mathbb{R}^{d \times n_x}$  and  $\mathbf{Y} \in \mathbb{R}^{d \times n_y}$ , dynamic time warping is a technique to optimally align samples of  $\mathbf{X}$  and  $\mathbf{Y}$  such that the following sum-of-squares is minimized:

$$J_{\text{DTW}}(\mathbf{P}) = \sum_{t=1}^M \|\mathbf{x}_{p_t^x}^x - \mathbf{y}_{p_t^y}^y\|^2, \quad (4.2)$$

where  $m$  is the number of steps needed to align both the signals,  $\mathbf{P} = [\mathbf{p}^x, \mathbf{p}^y]^T \in \mathbb{R}^{2 \times m}$  is the correspondance matrix in which  $\mathbf{p}^x \in \{1 : n_x\}^{m \times 1}$  and  $\mathbf{p}^y \in \{1 : n_y\}^{m \times 1}$  denote the composition of alignment in frames. For example, the  $i$ th frame in  $\mathbf{X}$  and the  $j$ th frame in  $\mathbf{Y}$  are aligned if  $\mathbf{p}_t = [p_t^x, p_t^y]^T = [i, j]^T$  for some  $t$ . Also,  $\mathbf{P}$  has to satisfy three constraints: (1) boundary condition  $\mathbf{p}_1 \equiv [1, 1]^T$  and  $\mathbf{p}_m \equiv [n_x, n_y]^T$ , (2) continuity ( $\mathbf{0} \leq \mathbf{p}_t - \mathbf{p}_{t-1} \leq \mathbf{1}$ ), and (3) monotonicity ( $t_1 \geq t_2 \Rightarrow \mathbf{p}_{t_1} - \mathbf{p}_{t_2} \geq \mathbf{0}$ ).

The number of possible ways to align  $\mathbf{X}$  and  $\mathbf{Y}$  is exponential in  $n_x$  and  $n_y$ , but luckily *dynamic programming* offers an efficient  $O(n_x n_y)$  approach to minimize  $J_{\text{DTW}}$  using Bellman's equation [7]:

$$L^*(\mathbf{p}_t) = \min_{\pi_t} \|\mathbf{x}_{p_t^x}^x - \mathbf{y}_{p_t^y}^y\|^2 + L^*(\mathbf{p}_{t+1}), \quad (4.3)$$

where the cost-to-go function  $L^*(\mathbf{p}_t)$  represents the remaining cost starting at  $t$ th step to be incurred following the optimum policy  $\pi^*$ . The policy function  $\pi : \{1 : n_x\} \times \{1 : n_y\} \rightarrow \{[1, 0]^T, [0, 1]^T, [1, 1]^T\}$  defines the transition between steps consecutive steps:  $\mathbf{p}_{t+1} = \mathbf{p}_t + \pi(\mathbf{p}_t)$ . Once the policy function is known, the alignment steps can be recursively constructed from starting point  $\mathbf{p}_1 = [1, 1]^T$ . [112]

### 4.1.3 Canonical Time Warping

Canonical time warping (CTW) [112] can be viewed as an extension of canonical correlation analysis (CCA) for spatio-temporal alignment of human motion between two subjects. CTW combines CCA with dynamic time warping (DTW) and extends CCA by allowing local spatial deformations.

In order to have a compact energy function for CTW, it is important to rewrite the Equation (4.2) as

$$J_{\text{DTW}}(\mathbf{W}_x, \mathbf{W}_y) = \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \mathbf{w}_i^{xT} \mathbf{w}_j^y \|\mathbf{x}_i - \mathbf{y}_j\|^2 = \|\mathbf{X}\mathbf{W}_x^T - \mathbf{Y}\mathbf{W}_y^T\|_F^2, \quad (4.4)$$

where  $\mathbf{W}_x \in \{0, 1\}^{m \times n_x}$  and  $\mathbf{W}_y \in \{0, 1\}^{m \times n_y}$  are binary selection matrices that need to be inferred to align  $\mathbf{X}$  and  $\mathbf{Y}$ . In the above Equation (4.4) the matrices  $\mathbf{W}_x$  and  $\mathbf{W}_y$  encode the alignment path. Observe that the Equation (4.4) is very similar to CCA's objective function (4.1). CCA applies a linear transformation to the rows (features) while DTW applies binary transformations to the columns (time).

The main idea of CTW is to add a linear transformation ( $\mathbf{V}_x^T, \mathbf{V}_y^T$ ) to the least-squares form of DTW in Equation (4.4):

$$J_{\text{CTW}}(\mathbf{W}_x, \mathbf{W}_y, \mathbf{Y}_x, \mathbf{Y}_y) = \|\mathbf{V}_x^T \mathbf{X} \mathbf{W}_x^T - \mathbf{V}_y^T \mathbf{Y} \mathbf{W}_y^T\|_F^2, \quad (4.5)$$

where  $\mathbf{V}_x \in \mathbb{R}^{d_x \times b}$ ,  $\mathbf{V}_y \in \mathbb{R}^{d_y \times b}$ ,  $b \leq \min(d_x, d_y)$  parameterize the spatial warping by projecting the sequences into a same coordinate system, while  $\mathbf{W}_x$  and  $\mathbf{W}_y$  warp the signals in time to achieve an optimal temporal alignment. This linear transformation allows differences in style and subject variability, adds a feature selection mechanism, and reduces the dimensionality of the signals. Similar to CCA, to make CTW invariant to translation, rotation, and scaling, it has the following constraints: (1)  $\mathbf{X} \mathbf{W}_x^T \mathbf{1}_m = \mathbf{0}_{d_x}$ ,  $\mathbf{Y} \mathbf{W}_y^T \mathbf{1}_m = \mathbf{0}_{d_y}$ , (2)  $\mathbf{V}_x^T \mathbf{X} \mathbf{W}_x^T \mathbf{W}_x \mathbf{X}^T \mathbf{V}_x = \mathbf{V}_y^T \mathbf{Y} \mathbf{W}_y^T \mathbf{W}_y \mathbf{Y}^T \mathbf{V}_y = \mathbf{I}_b$ , and (3)  $\mathbf{V}_x^T \mathbf{X} \mathbf{W}_y \mathbf{Y}^T \mathbf{V}_y$  to be a diagonal matrix.

The optimization of  $J_{\text{CTW}}$  is a non-convex optimization problem with respect to the alignment matrices ( $\mathbf{W}_x, \mathbf{W}_y$ ) and projection matrices ( $\mathbf{Y}_x, \mathbf{Y}_y$ ). It is possible to alternate between each step by first solving for ( $\mathbf{W}_x, \mathbf{W}_y$ ) using DTW, and then computing the optimal spatial projections ( $\mathbf{Y}_x, \mathbf{Y}_y$ ) by CCA. The steps monotonically decrease  $J_{\text{CTW}}$  and since the function is bounded below it will converge to a critical point.

The algorithm is discussed more in depth in the original work by Zhou and De la Torre [112]. It is also possible to derive a local version of the CTW algorithm by allowing multiple local spatial deformations with the introduction of weighting matrices and some regularization terms. In this thesis, the basic (global) CTW algorithm was effectively used to align the motions in time to get rid off slight temporal variations in the repetitions of a motion pattern.

## 4.2 Dimension Reduction by Machine Learning

Machine learning methods are generally split into three categories: supervised learning, reinforcement learning, and unsupervised learning. In *supervised learning* the data is split into inputs and outputs, and a certain classifier is learned with the training data. In *reinforcement learning*, a reward is associated with achieving a set goal and the learning agent tries to choose the actions that will maximize the cumulative reward. In *unsupervised learning* the objective is to understand the hidden structure of a data set, which is often unlabeled. In unsupervised learning, there is no error or reward signal to evaluate a potential solution, which separates unsupervised learning from supervised and reinforcement learning.

In this work, we are interested in learning the primitive features of the motion data space, such that the dimensionality of the original data is reduced but no motion information of importance would be lost. The underlying assumption is that the true data “lives” on a lower-dimensional manifold or surface embedded in the higher-dimensional, original vector space or observation space. By representing the original data on this manifold one can reduce the dimensionality of the data.

In general, the objective of dimension reduction is to find a mapping from the original  $d$ -dimensional space to a  $k$ -dimensional space where  $k < d$ . In this task, the methods of unsupervised learning are central. Based on the properties of this mapping, the methods can be divided into linear and nonlinear. In linear methods, the lower-dimensional manifold is restricted to a linear subspace. Nonlinear mapping methods extend the set of possible subspaces to nonlinear manifolds.

In this thesis, two dimension reduction methods, or subspace methods, are used: Principal Component Analysis (PCA) and Gaussian Processes Latent Variable Model (GP-LVM). PCA is an old and well-known dimension reduction technique which has been extended into a probabilistic and a Bayesian model. GP-LVM is a quite novel technique, which can interpretate PCA as a linear Gaussian process prior on a mapping from a latent space to the observed data space, but in general is not restricted to linear mappings and extends the dimension reduction to nonlinear subspaces.

### 4.2.1 Principal Component Analysis

Principal component analysis (PCA) is a technique that is widely used for applications such as dimension reduction, (lossy) data compression, feature extraction, and data visualization. It is also known as the *Karhunen-Loève*



*transform* [8]. As such, PCA is an example of dimension reduction techniques where the task is to find a mapping from the original  $d$ -dimensional space to a  $k$ -dimensional subspace where  $k < d$ .

There are two common definitions of PCA that give rise to the same algorithm. PCA can be defined as the orthogonal projection of the data onto a lower-dimensional linear space, the *principal subspace*, such that the variance of the projected data is maximized, that is, the maximum amount of information is preserved [40]. Equivalently, and originally, PCA can be defined as the linear projection that minimizes the average projection cost, which is defined as the mean squared distance between the data points and their projections [76].

In the maximum variance formulation, PCA can be done by *eigenvalue decomposition* of a data covariance matrix or *singular value decomposition* (SVD) of a data matrix  $\mathbf{X}$ , usually after mean-centering and normalizing the data matrix for each attribute. Nowadays, efficient algorithms exist to calculate the SVD of  $\mathbf{X}$ , so the SVD method is now the standard way to compute a PCA from the data, unless only a handful of principal components are required.

It can be shown that the optimal projection into a  $k$ -dimensional subspace is such that we choose  $k$  eigenvectors  $\mathbf{w}_j$ ,  $j = 1, \dots, k$ , of the data covariance matrix  $\mathbf{S} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T$  corresponding to the  $k$  largest eigenvalues  $\lambda_1, \dots, \lambda_k$ . Then a linear transformation of a data vector  $\mathbf{x}_n$  onto the principal subspace defined by the  $k$  eigenvectors is simply

$$\mathbf{z}_n = \mathbf{W}^T(\mathbf{x}_n - \boldsymbol{\mu}), \quad (4.6)$$

where  $\mathbf{z}_n$  are called the z-scores for the data vector, the  $k$  columns of  $\mathbf{W}$  are the  $k$  largest eigenvectors of  $\mathbf{S}$  and  $\boldsymbol{\mu}$  is the mean, or the bias, of the data.

### 4.2.2 Probabilistic PCA

One limitation of the conventional PCA is that it does not define a probability distribution. The reformulation of PCA as a probabilistic model was first proposed independently by Tipping and Bishop [96] and Roweis [83]. It has several advantages over the standard PCA, for example, allowing an *EM algorithm* for PCA, the possibility to deal with *missing values*, and forming a basis for Bayesian PCA [8]. Specifically, the  $d$ -dimensional observed variable  $\mathbf{x}$  is defined by a linear transformation of the  $k$ -dimensional latent variable  $\mathbf{z}$ , plus additive Gaussian noise:

$$\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \boldsymbol{\mu} + \boldsymbol{\epsilon}_n. \quad (4.7)$$

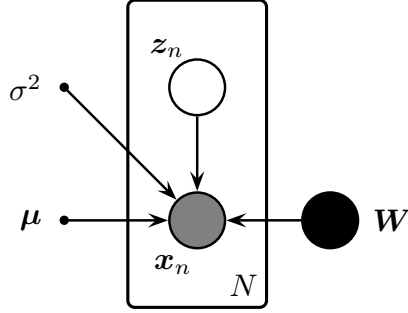


Figure 4.1: The plate diagram of PPCA.

It was shown that the *maximum likelihood* (ML) solution of the aforementioned model 4.7 extracts the principal components of the data. This model is illustrated in the plate diagram in Figure 4.1. The nodes represent random variables and the connecting edges represent their relationships. *Grey* nodes correspond to observed variables, *white* nodes correspond to unobserved or hidden variables (which are marginalized), *black* nodes correspond to variables to be optimised. Deterministic parameters are shown explicitly by small solid nodes. The plate, box labeled  $N$ , implicates that there are  $N$  observations of variables  $\mathbf{z}$  and  $\mathbf{x}$ .

Note that this framework is based on a mapping from a latent space to the data space, in contrast to the more conventional view of PCA. However, the reverse mapping from the data space to a latent space can be easily obtained using Bayes' theorem. The common choice of an isotropic Gaussian noise model  $\epsilon_n \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$  leads to the probability distribution

$$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2 \mathbf{I}). \quad (4.8)$$

The marginal distribution of  $\mathbf{x}$ , or the *predictive distribution*, is likewise Gaussian

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C}), \quad (4.9)$$

where  $\mathbf{C} = \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I}$ .

As well as the predictive distribution  $p(\mathbf{x})$  we also require the *posterior distribution*  $p(\mathbf{z}|\mathbf{x})$  which can be written as follows:

$$p(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}|\mathbf{M}^{-1}\mathbf{W}^T(\mathbf{x} - \boldsymbol{\mu}), \sigma^2 \mathbf{M}^{-1}), \quad (4.10)$$

where the  $k \times k$  matrix  $\mathbf{M}$  is defined by  $\mathbf{M} = \mathbf{W}^T \mathbf{W} + \sigma^2 \mathbf{I}$ .

According to the predictive distribution (4.9), the marginal likelihood for each data point is

$$p(\mathbf{x}_n|\boldsymbol{\mu}, \mathbf{W}, \sigma^2) = \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I}). \quad (4.11)$$

Taking advantage of the independence of the data points, the log-likelihood of the full data set is

$$\log \mathcal{L}(\mathbf{X}) = \log p(\mathbf{X}|\boldsymbol{\mu}, \mathbf{W}, \sigma^2) = \sum_{n=1}^N \log p(\mathbf{x}_n|\mathbf{W}, \boldsymbol{\mu}, \sigma^2). \quad (4.12)$$

Tipping and Bishop [96] showed that the analytic solution to the maximization of the likelihood can be obtained in closed form, and the *maximum* is obtained when the  $k$  eigenvectors are chosen such that they have the  $k$  largest eigenvalues. If eigenvectors of the ML solution  $\mathbf{M}_{\text{ML}}$  are arranged in decreasing order by their corresponding eigenvalues, the columns (eigenvectors) of  $\mathbf{M}_{\text{ML}}$  define the principal subspace of standard PCA. Therefore, this model has an interpretation as a *probabilistic* version of PCA.

Lawrence [51] showed that instead of optimizing parameters and marginalising latent variables, we can marginalize the parameters  $\mathbf{W}$ , treating them as random variables, and optimize with respect to latent variables  $\mathbf{Z} = \{\mathbf{z}_n\}_n^N$ . For a particular choice of prior distribution on  $\mathbf{W}$ , this probabilistic model will also turn out to be equivalent to PCA. However, the resulting marginalised likelihood  $p(\mathbf{X}|\mathbf{Z}, \sigma^2\mathbf{I})$  can be interpreted as a product of  $d$  independent *Gaussian processes*. Then, by replacing the linear Gaussian process prior over the space of functions (the linear covariance function, or linear inner-product kernel) with a nonlinear covariance function, we can obtain nonlinear mappings from the latent space to the observed space. Thus, we have a nonlinear latent variable model, titled as the Gaussian process latent variable model, which will be discussed in the next Section 4.2.3.

### 4.2.3 Gaussian Process Latent Variable Model

Gaussian processes [75, 103, 104] are a class of probabilistic models which can describe distributions over function spaces. Formally, a Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution [81]. In general, a Gaussian process is a stochastic process which is completely specified by its mean function and covariance function. We define the mean function  $m(\mathbf{x})$  and the covariance function  $k(\mathbf{x}, \mathbf{x}')$  of a real process  $f(\mathbf{x})$  as

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})] \quad (4.13)$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] \quad (4.14)$$

and write the Gaussian process (GP) as

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \quad (4.15)$$

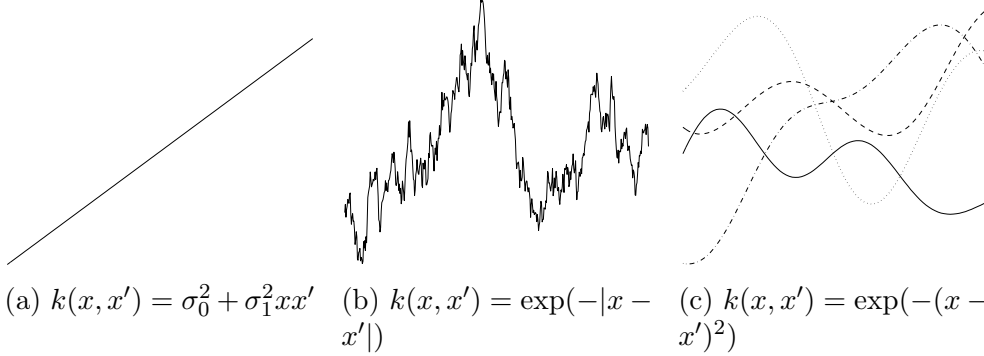


Figure 4.2: The effect of the choice of a covariance function  $k(x, x')$  on a Gaussian process. Adapted from [102].

For notation simplicity and to simplify calculations, the mean function is usually taken to be zero, without the loss of generality, while the covariance function is necessarily constrained to produce positive definitive matrices.

In the traditional probabilistic inference framework, we are given a set of input-output pairs, in matrices  $\mathbf{Z} \in \mathbb{R}^{N \times k}$  and  $\mathbf{X} \in \mathbb{R}^{N \times d}$  respectively, and seek to estimate the unobserved *latent* function  $f(\mathbf{z})$ , responsible for generating  $\mathbf{X}$  given  $\mathbf{Z}$ . In this *supervised learning* setting, Gaussian processes can be employed as nonparametric prior distributions over the latent function  $f(\mathbf{z})$ . Each data point  $\mathbf{x}_n$  is generated from the corresponding  $f(\mathbf{z}_n)$  by adding independent Gaussian noise,

$$\mathbf{x}_n = f(\mathbf{z}_n) + \boldsymbol{\epsilon}_n, \quad \boldsymbol{\epsilon}_n \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}), \quad (4.16)$$

where  $f$  is drawn from a Gaussian process,  $f(\mathbf{z}_n) \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{z}, \mathbf{z}'))$ . This GP prior only depends on the covariance function  $k$ , or kernel, operating on the inputs  $\mathbf{Z}$ . As the intention is to obtain a flexible model, only general assumptions are made about the form of the generative mapping  $f$  which is reflected in the choice of the covariance function. For example, an exponential quadratic covariance function, also known as radial basis function (RBF),

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_{se}^2 \exp \left( -\frac{(\mathbf{x}_i - \mathbf{x}_j)^2}{2l^2} \right), \quad (4.17)$$

forces the latent functions to be infinitely smooth. The effect of the choice of a covariance function on 1-D Gaussian processes is illustrated in the Figure 4.2.

The collection of latent function instantiations, denoted by  $\mathbf{F} = \{\mathbf{f}_n\}_n^N$ , is normally distributed which allows us to compute analytically the marginal

likelihood

$$p(\mathbf{X}|\mathbf{Z}) = \int \prod_{n=1}^N p(\mathbf{x}_n|\mathbf{f}_n)p(\mathbf{f}_n|\mathbf{z}_n)d\mathbf{F} \quad (4.18)$$

$$= \mathcal{N}(\mathbf{X}|\mathbf{0}, \mathbf{K}_{NN} + \sigma_\epsilon^2 \mathbf{I}), \quad (4.19)$$

where  $\mathbf{K}_{NN} = k(\mathbf{Z}, \mathbf{Z})$ . We can then view this model as an *unsupervised learning* scenario, where the input data  $\mathbf{Z}$  are not directly observed and the purpose is to do dimension reduction,  $k \ll D$ . The *Gaussian process latent variable model* (GP-LVM) [50, 51] provides a solution to this problem by treating the unobserved inputs  $\mathbf{Z}$  as latent variables while employing a product of  $D$  independent GPs as a prior for the latent mapping. The assumed generative model takes the form

$$x_{nd} = f_d(\mathbf{z}_n) + \epsilon_{nd}, \quad (4.20)$$

where  $\epsilon_{nd}$  is zero-mean Gaussian noise with variance  $\sigma_\epsilon^2$  and  $\mathbf{F} = \{\mathbf{f}_d\}_{d=1}^D$  with  $f_{nd} = f_d(\mathbf{z}_n)$ . Given a finite data set the GP priors take the form

$$p(\mathbf{F}|\mathbf{Z}) = \prod_{d=1}^D \mathcal{N}(\mathbf{f}_d|\mathbf{0}, \mathbf{K}_{NN}) \quad (4.21)$$

which is normally distributed and thus allows for general nonlinear mappings to be marginalized out analytically, to obtain the likelihood

$$p(\mathbf{X}|\mathbf{Z}) = \prod_{d=1}^D \mathcal{N}(\mathbf{x}_d|\mathbf{0}, \mathbf{K}_{NN} + \sigma_\epsilon^2 \mathbf{I}), \quad (4.22)$$

where  $\mathbf{x}_d$  is the  $d$ th column of  $\mathbf{X}$ , and which is analogous to Equation (4.19). Figure 4.3 shows the Gaussian process as a latent variable model represented as a plate diagram (see Section 4.2.2 for description).

#### 4.2.4 Learning in the GP-LVM

Learning in the GP-LVM consists of maximising the likelihood (4.22) with respect to the positions of the latent variables  $\mathbf{Z}$  and the parameters  $\boldsymbol{\theta}$  of the kernel. This leads to the following log-likelihood function

$$\log \mathcal{L}(\mathbf{Z}, \boldsymbol{\theta}) = -\frac{dN}{2} \log 2\pi - \underbrace{\frac{d}{2} \log |\mathbf{K}|}_{\text{complexity}} \underbrace{-\frac{1}{2} \text{tr}(\mathbf{K}^{-1} \mathbf{X} \mathbf{X}^T)}_{\text{data-fit}}, \quad (4.23)$$

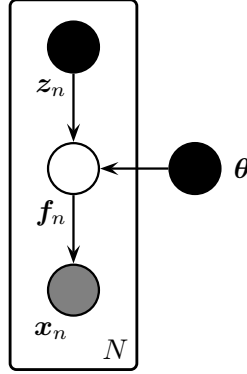


Figure 4.3: The plate diagram of GP-LVM.

which shows two “competing” terms, the data-fit and the complexity term. The complexity term measures and penalizes for the complexity of the model, while the data-fit term measures how well the model fits the data. This “competition” encourages the GP-LVM model not to over-fit the data.

Gradients of  $\mathcal{L}(\mathbf{Z}, \boldsymbol{\theta})$  are then obtained through combining gradients of  $\frac{\partial \mathcal{L}(\mathbf{Z}, \boldsymbol{\theta})}{\partial \mathbf{K}}$  with gradients given by  $\frac{\partial \mathbf{K}}{\partial \mathbf{Z}}$  and  $\frac{\partial \mathbf{K}}{\partial \boldsymbol{\theta}}$ . In general, it is not possible to obtain a fixed point solution for  $\mathbf{X}$  and  $\boldsymbol{\theta}$  (unless the kernel  $\mathbf{K}$  is linear, thus resulting into PPCA) and to make progress gradient-based iterative optimisation has to be used. These methods have usually high computational cost because such algorithms rely on multiple re-evaluations of the log-likelihood and its gradients, and each evaluation has  $O(N^3)$  complexity due to the inverse of  $\mathbf{K}$ . Luckily, there exists multiple sparse approximation algorithms in GP regression literature, and the computations can be speeded up ( $O(k^2 N)$ ) in an expanded probability model where the GP prior is augmented to include auxiliary inducing variables [13, 20, 21, 84, 92, 97].

Extending the variational sparse GP method of Titsias [97], Titsias and Lawrence [98] showed that a closed form Jensen’s lower bound of the GP-LVM marginal likelihood can be computed, allowing variational inference in the GP-LVM which makes full Bayesian training of the model possible which is robust to overfitting. Furthermore, by using *automatic relevance determination* (ARD) [54, 70] squared exponent kernel (RBF), the algorithm allows an automatic inference of the dimensionality of the nonlinear latent space, and can also deal with missing or uncertain inputs [98].

In this thesis, Bayesian variational GP-LVM has been effectively used to find the latent nonlinear subspace of observed MOCAP data with relatively high dimensionality which may include large amounts of redundant information.

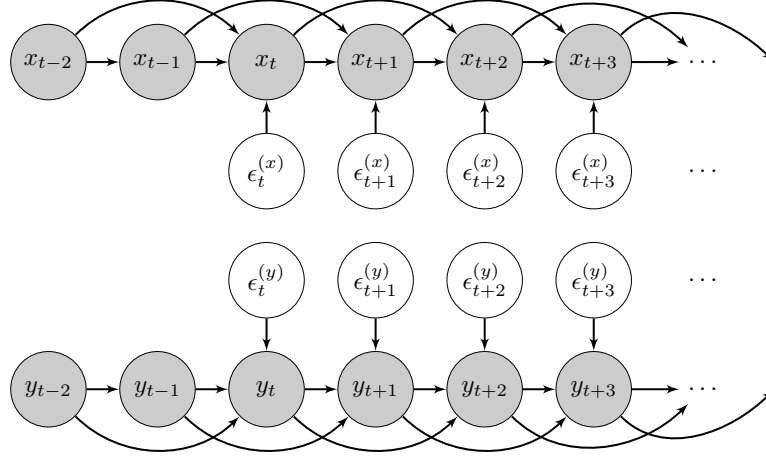


Figure 4.4: The second-order autoregressive models.

## 4.3 Information Estimation

### 4.3.1 Complexity Estimation

In statistics and signal processing, an autoregressive (AR) model [11] is a type of a stochastic time-varying process, where the output variable depends linearly on its own previous values. In the general form, an autoregressive model of order  $p$ ,  $\text{AR}(p)$ , is defined as

$$x_t = \beta_0 + \sum_{i=1}^p \beta_i x_{t-i} + \epsilon_t, \quad (4.24)$$

where  $\beta_0, \dots, \beta_p$  are the parameters of the model ( $\beta_0$  is a constant) and  $\epsilon_t$  is white noise, usually a zero-mean, independent and identically distributed Gaussian variate.

The movement in motion sequences  $X$  and  $Y$  is modelled as a second-order autoregressive model ( $\text{AR}(2)$ ):

$$x_t = \beta_0 + \beta_1 x_{t-1} + \beta_2 x_{t-2} + \epsilon_t^{(x)}, \quad (4.25a)$$

$$y_t = \eta_0 + \eta_1 y_{t-1} + \eta_2 y_{t-2} + \epsilon_t^{(y)}, \quad (4.25b)$$

where  $\beta_0, \beta_1, \beta_2$ , and  $\eta_0, \eta_1, \eta_2$  are real-valued parameters to be fitted via least-squares problem. The appeal of this simple autoregressive model is its simplicity and interpretability. In a simple system, the predictable changes of the time series are autocorrelated, and any additional information is needed to diverge from the basic state. The errors  $\epsilon^{(x)}$  and  $\epsilon^{(y)}$  are assumed to

be independent zero-mean Gaussian variates, and can be interpreted as innovations made in the motion. Since the two sequences are supposed to be instances of the same motion pattern, they will typically be correlated. The model is represented in the Figure 4.4.

After the parameter fitting, we obtain the residuals

$$r_t^{(x)} = x_t - \hat{x}_t = x_t - (\hat{\beta}_0 + \hat{\beta}_1 x_{t-1} + \hat{\beta}_2 x_{t-2}), \quad (4.26a)$$

$$r_t^{(y)} = y_t - \hat{y}_t = y_t - (\hat{\eta}_0 + \hat{\eta}_1 y_{t-1} + \hat{\eta}_2 y_{t-2}), \quad (4.26b)$$

where  $\hat{x}_t$  and  $\hat{y}_t$  denote the predictions obtained by the AR2 model. Under the model (4.25), the residuals can be thought as approximations of the “innovations”. Thus, this formulation captures the complexity as the amount of changes in the trajectory. The total complexity  $r_{\text{tot}} = \{r_t\}_t^N$  of a sequence can then be viewed as the *entropy* of a certain motion, discussed next.

### 4.3.2 Entropy

In information theory, *entropy* is a measure of the uncertainty in a random variable and it quantifies the expected value of the information contained in a message. In this context, it is also referred as *Shannon’s entropy* in contrast to entropy in thermodynamics, which was first introduced by Rudolf Clausius [16] in 1865 and later refined as a statistical measure using probability theory by Ludwig Boltzmann [10] in 1877. Shannon’s entropy formula shares a close resemblance to the statistical thermodynamic entropy formulation called *Gibbs entropy*, hence the inspiration for the name also in information theory.

The information theoretical definition of entropy was introduced by Claude E. Shannon [85] in his infamous paper *A Mathematical Theory of Communication* in 1948. Shannon defined the entropy  $H$  of a discrete random variable  $X$  with possible values  $x_1, \dots, x_n$  and a probability mass function  $p_X(x)$  as:

$$H(X) = - \sum_i p_X(x_i) \log_b p_X(x_i), \quad (4.27)$$

where  $b$  is the base of the logarithm used. Common values of  $b$  are 2 or Euler’s number  $e$ , and the respective units for entropy are *bit* or *nat*. Unless otherwise specified, all logarithms will be in base 2 and hence all entropies will be in bits.

For example, the entropy of a fair coin toss is 1 bit. When the coin is fair, the entropy of the coin toss is as high as it could be. This means that there is no way to predict the outcome, and learning the outcome contains 1 bit



of information, which means that the minimum expected number of binary questions required to determine the outcome is one.

When the discrete entropy formulation is extended to the continuous case, it is called *differential entropy*. The differential entropy  $h$  of a continuous random variable  $X$  with a probability density function  $f(x)$  is defined as

$$h(X) = - \int_S f(x) \log f(x) dx, \quad (4.28)$$

where  $S$  is the *support set* of the random variable, i.e. the set where  $f(x) > 0$ . As with all probability densities, one must be careful to construct examples where the integral exists. Also,  $f(x) \log f(x)$  is understood to be zero whenever  $f(x) = 0$ . Unlike discrete entropy, differential entropy can also be negative.

For zero-mean normal distribution, if  $X \sim f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$ , the differential entropy in nats is:

$$\begin{aligned} h(f) &= - \int_{-\infty}^{\infty} f(x) \ln f(x) dx \\ &= - \int_{-\infty}^{\infty} f(x) \left( -\frac{x^2}{2\sigma^2} - \ln \sqrt{2\pi\sigma^2} \right) \\ &= \frac{1}{2\sigma^2} \int_{-\infty}^{\infty} x^2 f(x) dx + \ln(2\pi\sigma^2)^{\frac{1}{2}} \int_{-\infty}^{\infty} f(x) dx \\ &= \frac{1}{2\sigma^2} \mathbb{E}[X^2] + \ln(2\pi\sigma^2)^{\frac{1}{2}} \\ &= \frac{1}{2\sigma^2} \sigma^2 + \frac{1}{2} \ln 2\pi\sigma^2 \\ &= \frac{1}{2} \ln e + \frac{1}{2} \ln 2\pi\sigma^2 \\ &= \frac{1}{2} \ln 2\pi e \sigma^2 \quad \text{nats.} \end{aligned}$$

Changing the base of the logarithm, the differential entropy of a zero-mean normal distribution is

$$h(f) = \frac{1}{2} \log 2\pi e \sigma^2 \quad \text{bits.} \quad (4.29)$$

Therefore, the entropy of a normal distribution is determined by its variance.

In general, the entropy of a multivariate normal distribution,  $f \sim \mathcal{N}_n(\mu, \Sigma)$ , is [3, 18]:

$$h(f) = - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(\mathbf{x}) \log f(\mathbf{x}) d\mathbf{x} = \frac{1}{2} \log [(2\pi e)^n |\Sigma|]. \quad (4.30)$$

For a bivariate normal distribution,  $(X, Y) \sim \mathcal{N}_2(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ , the covariance matrix is:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} \quad (4.31)$$

Therefore, when we take the determinant of (4.31), the entropy of a bivariate normal distribution is:

$$h(f(x, y)) = \frac{1}{2} \log [(2\pi e)^2 (\sigma_x^2 \sigma_y^2 - V_{xy}^2)], \quad (4.32)$$

where  $V_{xy}$  is the covariance of  $X$  and  $Y$ .

### 4.3.3 Relative Entropy

The *relative entropy* is an asymmetric measure of the difference between two probability distributions, also called as *Kullback–Leibler divergence* or distance. Specifically, the relative entropy  $D(p||q)$ , divergence of  $q$  from  $p$ , is a measure of the inefficiency, or the information lost, when assuming that the distribution is  $q$  when the true distribution is  $p$ . Typically,  $p$  represents the “true” distribution or the best observations of the data, when  $q$  represents an approximation or a descriptive model of the data.

For discrete cases, the relative entropy between two probability mass functions  $p$  and  $q$  is defined as:

$$D(p||q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}. \quad (4.33)$$

Basically, it is the expectation of the logarithm of the likelihood ratio between  $p$  and  $q$  where the expectation is evaluated using probabilities  $p(x)$ . It is only defined if  $p$  and  $q$  both sum to 1, and by convention that  $0 \log \frac{0}{q} = 0$  and  $p \log \frac{p}{0} = \infty$ .

For continuous cases, the relative entropy between two probability density functions  $f$  and  $g$  is defined as:

$$D(f||g) = \int_S f(x) \log \frac{f(x)}{g(x)} dx. \quad (4.34)$$

It is finite only if the support set  $S$  of  $f(x)$  is contained in the support set of  $g(x)$ .

### 4.3.4 Mutual Information

If we consider two discrete random variables  $X$  and  $Y$  with a joint probability mass function  $p(x, y)$  and marginal probability mass functions  $p(x)$  and  $p(y)$ ,

then the *mutual information*  $I(X; Y)$  is the relative entropy between the joint distribution and the product distribution  $p(x)p(y)$ :

$$I(X; Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (4.35)$$

$$= D(p(x, y) || p(x)p(y)) \quad (4.36)$$

$$= \mathbb{E}_{p(x, y)} \left[ \log \frac{p(X, Y)}{p(X)p(Y)} \right] \quad (4.37)$$

$$(4.38)$$

Mutual information is a measure of the amount of information that one random variable contains about the another random variable. In other words, it is the reduction in the uncertainty of one random variable due to knowledge of the other. This can be shown by rewriting the definition as follows:

$$I(X; Y) = \sum_{x, y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (4.39)$$

$$= \sum_{x, y} p(x, y) \log \frac{p(x|y)}{p(x)} \quad (4.40)$$

$$= - \sum_{x, y} p(x, y) \log p(x) + \sum_{x, y} p(x, y) \log p(x|y) \quad (4.41)$$

$$= - \sum_{x, y} p(x, y) \log p(x) + \sum_{x, y} p(x, y) \log p(x|y) \quad (4.42)$$

$$= H(X) - H(X|Y) \quad (4.43)$$

The relationship between  $H(X)$ ,  $H(Y)$ ,  $H(X, Y)$ ,  $H(X|Y)$ ,  $H(Y|X)$  and  $I(X, Y)$  is represented in a Venn diagram in Figure 4.5. Basically, the mutual information  $I(X; Y)$  corresponds to the intersection of the information in  $X$  with the information in  $Y$ .

From the definition (4.43) it is clear that the mutual information, in contrast to the relative entropy, is symmetrical:

$$I(X; Y) = H(Y) - H(Y|X) = H(X) - H(X|Y). \quad (4.44)$$

The mutual information  $I(X; Y)$  between two continuous random variables with a joint density  $f(x, y)$  is defined as

$$I(X; Y) = \int f(x, y) \log \frac{f(x, y)}{f(x)f(y)} dx dy. \quad (4.45)$$

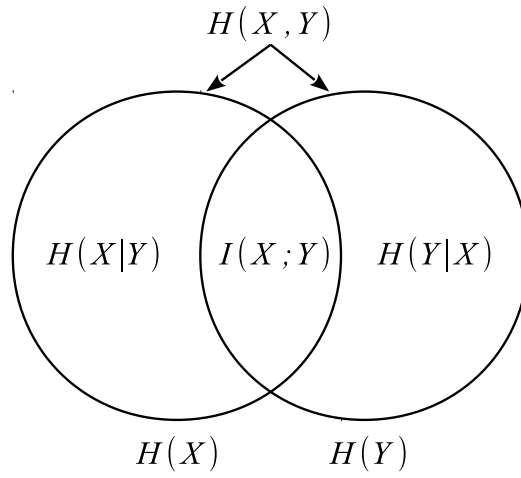


Figure 4.5: The relationship between mutual information and entropy represented in a Venn diagram.

Finally, we can also note that

$$I(X; X) = H(X) - H(X|X) = H(X). \quad (4.46)$$

This means that the mutual information of a random variable with itself is the entropy of the random variable. Thus, the entropy is sometimes referred to as *self-information*.

### 4.3.5 Mutual Information of a Bivariate Normal Distribution

By using the results of (4.29) and (4.32), the mutual information of a bivariate normal distribution is:

$$\begin{aligned}
 I(X; Y) &= H(X) + H(Y) - H(X, Y) \\
 &= \frac{1}{2} \log(2\pi e \sigma_x^2) + \frac{1}{2} \log(2\pi e \sigma_y^2) - \frac{1}{2} \log[(2\pi e)^2 (\sigma_x^2 \sigma_y^2 - V_{xy}^2)] \\
 &= \frac{1}{2} \log\left(\frac{\sigma_x^2 \sigma_y^2}{\sigma_x^2 \sigma_y^2 - V_{xy}^2}\right) \\
 &= -\frac{1}{2} \log\left(\frac{\sigma_y^2 - V_{xy}^2 / \sigma_x^2}{\sigma_y^2}\right) \\
 &= -\frac{1}{2} \log\left(1 - \left(\frac{V_{xy}}{\sigma_x \sigma_y}\right)^2\right) \\
 &= -\frac{1}{2} \log(1 - \rho_{xy}^2), \tag{4.47}
 \end{aligned}$$

where  $\rho_{xy} = V_{xy}(\sigma_x \sigma_y)^{-1}$  is the Pearson correlation coefficient of  $X$  and  $Y$ . We can see that the mutual information of bivariate normal  $(X, Y)$  is now fully determined by their correlation.

However, since we do not in general know the true correlation coefficient  $\rho_{xy}$ , we need to estimate it from the data, which causes some statistical error. The statistical variation of mutual information estimate—obtained by plugging the empirical correlation coefficient  $\hat{\rho}$  into Equation (4.47)—can be characterized by means of results from classical likelihood ratio test theory.

When  $n$  is the amount of independent samples, it can be shown that in the case of  $\rho_{xy} = 0$  ( $X$  and  $Y$  are independent) the asymptotical distribution of the mutual information estimate is  $\frac{1}{2n} \chi_{(J-1)(K-1)}^2$ , where  $J$  and  $K$  are the degrees of freedom and  $n$  is the number of samples [12]. In the case of  $\rho_{xy} \neq 0$ , the distribution of  $\hat{\rho}$  is asymptotically normal, centered at the true value of  $\rho_{xy}$ .

Although we don't know in advance which of the two cases holds for each feature of the data sequences, we can have a *conservative estimate* of the mutual information  $I(X; Y)$ , when we know the expected value  $\mathbb{E}[\frac{1}{2} \chi_1^2] = 0.5$ . By subtracting the bias 0.5 nats (or  $0.5 \log_2 e$  bits) from each feature, the estimation for mutual information in bits is

$$\hat{I}(X; Y) = \frac{n}{2} \log_2(1 - \hat{\rho}^2) - \frac{1}{2} \log_2 e, \tag{4.48}$$

where the last term is the bias correction.

### 4.3.6 Throughput

In information theory and computer science, throughput is defined as the information processed or communicated in a given period of time. Therefore, throughput is a metric of bits per second (bps). In HCI, throughput is also defined as the measured rate of performance – in Fitts' law studies a metric of bits per second.

When dealing with high-dimensional motion data, after the dimension reduction step, each motion sequence is represented as a collection of principal feature vectors. When calculating the total throughput, first each feature of sequences  $X$  and  $Y$  is independently handled to compute the mutual information (4.48), and the total estimate  $\hat{I}_{\text{tot}}$  is the sum of the feature throughputs. The total throughput estimate of a motion sequence  $X$  conditioned on sequence  $Y$  is therefore

$$\text{TP}(X|Y) = \frac{R\hat{I}_{\text{tot}}(X;Y)}{n} = -\frac{R}{2}\log_2(1 - \hat{p}^2) - \frac{R}{2n}\log_2 e, \quad (4.49)$$

where  $R$  denotes the framerate (frames per second) and  $n$  is the amount of frames. Because we want to avoid the overestimation of TP caused by the duplicated frames of CTW process, the duplicated frames are removed with respect to the motion sequence  $X$ . This makes the TP metric asymmetrical,  $\text{TP}(X|Y) \neq \text{TP}(Y|X)$ , in the evaluation of the experiments.

## Chapter 5

# Implementation

### 5.1 Capturing of Motions and Gestures

#### 5.1.1 Vicon and PhaseSpace

Traditionally, motion capturing has been based on physical markers attached on the body of the performer, and capturing full-body motion has required multiple cameras with high accuracy. These optical systems are usually divided in systems based on *passive markers* and in systems based on *active markers*. Systems with passive markers use markers coated with retroreflective materials (reflecting light with minimal scattering), while systems with active markers typically use illuminated LEDs. Marker-based systems typically require that the performer wears a special suite made from e.g. spandex or lycra, to which markers are attached by velcro. The suite might hinder the motion, so during the dance study, no suite was used as the markers could be attached directly to the performers clothes or skin.

The study for ballerina dance sequences was captured using *Vicon* motion capture system. The system consisted of 12 *F-40* cameras with IR optical filters and IR LEDs, several processor units, a set of reflective dot markers, and *Vicon Nexus* software. The system was able to capture 3D trajectories in submillimeter accuracy with a steady framerate of 120 fps. Trajectories were considered to be of high quality, that with minimal interpolation provided by the Vicon software, no further filtering was needed.

The study for bimanual in-air gesturing (The *Minority Report* scenario) was captured using *PhaseSpace* motion capture system. It consisted of 12 *Impulse* cameras, a processor unit, and a set of active LED markers. The system was able to capture 3D trajectories with a steady framerate of 120 fps. Trajectories contained some errors and visible instrumental noise, so interpolation and smoothing were needed before analysis.

### 5.1.2 Mouse Traces

The study for cyclical tapping was captured using a standard optical mouse. Software for the test was custom-made using *Pygame* Python modules designed for writing video games [88]. The aimed framerate for capturing traces was 100 fps but due to technical shortcomings, the average delay between samples was 10.4 ms, which results to an average sample rate of 96.15 Hz. The traces acquired were considered to be of good quality, so no further preprocessing was needed before analysis.

### 5.1.3 Microsoft Kinect

*Kinect* is a motion sensing input device developed by Microsoft for the Xbox 360 video game console and Windows PCs. The device features an RGB camera, IR depth sensor and a multi-array microphone. The depth sensor consists of an IR laser projector combined with a monochrome CMOS sensor. The depth sensor works under any ambient light conditions. This 3D range camera technology was developed by PrimeSense.

Microsoft's official non-commercial Kinect SDK has Windows compatible PC drivers for Kinect device [66]. Several open source framework alternatives are also available, the primary being *OpenNI Framework* [72] and *Open-Kinect* (libfreenect) [9]. The main framework for the implementation needed to be cross-platform and have efficient skeleton tracking libraries, so the most suitable framework was OpenNI with NITE middleware provided by PrimeSense.

OpenNI and NITE are able to track 15 different joints in the body. The GUI for an interactive demo software was made with *Processing* programming language [31] which also has a simple OpenNI and NITE wrapper *simple-openni*. This allowed fast developping for an interactive demo with Kinect.

## 5.2 Preprocessing of the Data

MOCAP devices have always some instrumental noise that affects measurements, therefore some filtering as a preprocessing step is always needed for the raw data. In the MOCAP literature for kinematics, a Butterworth lowpass filter with a cut-off frequency of 6–9 Hz has been found simple yet effective [106, 107]. With several trials by judging the visible quality of MOCAP features, a cut-off frequency of 7 Hz was chosen for the raw data collected from the PhaseSpace system. The Vicon system natively incorporates very effective filtering, so no additional preprocessing was needed in that case.



Also, MOCAP markers are sometimes occluded from the cameras' view-points, especially in the case of motions where the limbs are close to the torso, or close to each other. Therefore there is usually some missing information. For short missing time frames, a piecewise cubic spline interpolation was used. If just a couple components had long intervals with missing information (10 frames or more with 120 fps) a straightforward linear interpolation was used. In a case of extensive missing information with multiple components, the whole motion sequence was deemed unusable.

Quite often there is missing information at the start or at the end of motion sequence data in one or several components. Then it is most practical to properly cut off all the time frames which had components with missing information. Also, frames that contain unnecessary information—when the motion under examination has not yet started or has ended—need to be cut away because including unnecessary frames naturally affects the throughput.

### 5.3 Algorithms

*Canonical time warping* (CTW) was used for aligning two motion sequences accurately in a spatio-temporal manner. CTW combines *canonical correlation analysis* (CCA) with *dynamic time warping* (DTW) and extends CCA by allowing local spatial deformations. Basically, CTW finds the temporal alignment that maximizes the spatial correlation between two motion samples (see Section 4.1.3).

Alignment indices provided by CTW were used to duplicate the frames in each of the pair sequences locally. In effect, CTW is used to provide “lag” in each of the sequences when needed. The MATLAB algorithm code was provided by Zhou [111]. For these experiments, CTW was run with the default parameters suitable for MOCAP from the example code.

*Gaussian Process Latent Variable Model* (GP-LVM) was used for the dimension reduction for motion data. Fully Bayesian GP-LVM provides automatic model selection for the amount of latent dimensions (*automatic relevance determination*, ARD) with an RBF-ARD kernel. The model also integrates time dynamics as a prior variable. The algorithm was run with 100 inducing points, 50 initialization iterations (with fixed  $\beta$ -parameter) and a maximum of 1000 optimization iterations. The extensive MATLAB software was provided by Lawrence [49].

Alternatively, for ease and speed, *Principal Component Analysis* (PCA) was used for dimension reduction (see Section 4.2.1). According to a common practise, the number of principal components was chosen to explain 90% of total variance.

The software for shared information and throughput computation were implemented in R code. *Rscript* provides convenient interface for running the computations directly from UNIX shell. The software is freely available on the project website: <http://infocapacity.hiit.fi>.

## 5.4 Server Implementation

A public Internet service for evaluating throughputs with a convenient GUI was implemented on a Linux Debian-based LAMP server (**L**inux, **A**pache, **M**ySQL, **P**HP). The PHP web site enables the user to input either two data files, a directory of paired gestures, a directory of repetitions, or a directory with sub-folders of repetitions of different gestures. The available optional procedures are an alignment of the pairs of data with CTW, a dimension reduction with either PCA or Bayesian GP-LVM, and throughput evaluation with different FPS values. The user then receives the results by an email.

The computation algorithms are implemented in MATLAB and R, with a shell script that executes the commands in the server. Because the service has to enable multiple users at the same time, there needs to be a job queue for distributing the computation tasks one at a time in correct order without throttling the server, especially in the case of computationally heavy GP-LVM.

There are multiple open source software for job scheduling with time-consuming tasks. Simplicity and PHP implementation in mind, *Beanstalk* [80] program was chosen for an appropriate work queue manager with *Pheanstalk* [4] as the PHP client. Because Beanstalk only handles the messages between processes, the message-listening worker process had to be daemonized. *Supervisor* [62] program was chosen to run the worker in the background. Both Beanstalk and Supervisor provide convenient interfaces for process management, event logging, and memory management in case of memory leaks. The web front-end of the service is in Figure 5.1.

← → ↻ infocapacity.hiit.fi/service.php

## Service for Calculating the Information Capacity of Full-Body Movements

Logged in as: **amodig**  
[Logout](#) or [Change password](#)

### Method 1

Send a pair of sequences:

File 1:  No file chosen

File 2:  No file chosen

### Method 2

Send a compressed directory (zip or rar):

No file chosen

Folder structure:

☐ Multi-repetition folder
 ☐ Pair-folder
 ☐ Multiple sub-folders  
 (NOTE: See documentation for details.)

### Options

FPS:

Alignment:

☒ CTW

Dimension Reduction:

☐ None
 ☐ PCA
 ☐ Var-GP-LVM

Work ID (optional):

Receive the results to the following e-mail address:

E-mail:

Debug: Save residuals ☐ Save PCA-sequences ☐

### Help

Send either a pair of sequence data or a compressed directory of data files. The directory can consist of (1) repetitions of a single motion sequence, or (2) pairs of sequences, or (3) sequence sub-folders consisting of repetitions.

Remember to check that the data files to be compared have equal amount of features (column components). Please use leading zeros for file names to maintain correct order.

Fill in the frame rate (FPS) of the data. If you want to align the sequences, choose CTW. Select dimension reduction method: Principal Component Analysis (PCA), Gaussian Processes Latent Variable Model (GP-LVM), or none. Note that GP-LVM may take many hours to compute.

Please fill in your e-mail address for receiving results.

[Documentation](#)

[Project Page](#)








Figure 5.1: The Internet service for calculating the information capacity.

## Chapter 6

# Results and Evaluation

### 6.1 Classical Ballet

We recorded a dance performance of a teacher of classical and romantic ballet with several years of experience. She was asked to select dances that would reflect high precision and complexity, both fast and slow. The repetitions were to be as precise as possible both temporally and spatially. The motions could be repeated as many times as desired until she was satisfied with the quality of control. Typically just one repetition was needed if there was no technical errors related with the MOCAP system, which reflected her high level of experience. For the final analysis, six motion sequences were chosen.

The throughputs for the dance sequences with 120 FPS are listed in Table 6.1. The table shows a range of TPs of 213 to 590 bps with GP-LVM. With a nonexpert evaluation, the sequences related with worst TPs involved slow movement and stopping in static postures. In contrast, the highest TPs were related with fast multi-limb motions, circling and jumping. The GP-LVM was fixed to 6 latent dimensions.

As Bayesian variational GP-LVM is very slow to compute, a second alternative data set was created by dropping the framerate to 60 FPS. With this data, the GP-LVM model used Bayesian ARD kernel to find the optimal amount of latent dimensions, to include at least 90% of the variance in the latent space. The results are shown in Table 6.2. In some cases, the model prunes the original 111 dimensions very heavily, which results into very low throughput values. The model’s behaviour is not well-adjusted, so the results are not as robust as with the PCA method.

The throughput values also depend on the sample rate, or FPS. With raw data, throughputs are on average 30% higher when the framerate is doubled from 60 FPS to 120 FPS, and 52% higher after PCA. Therefore, it

Table 6.1: The throughputs of the ballet data sequences with 120 FPS. All GP-LVM models had 6 latent dimensions.

Sequence	TP		TP (GP-LVM)	
	raw	PCA	CTW	No CTW
Tombé pas de bourrée, italian fouetté, piqué turn, jeté en tournant	4111	1317	590	516
Adagio (passé devant développé, arabesque, écarté devant pas de bourré)	222	114	399	330
Petit jeté (glissade jeté, ballotté, ballon, entrechat, assemblé)	3375	775	314	289
Petit jeté (temps de cuisse, sissonne devant fermée, derrière fermée, sissonne ouvert pas de bourrée)	3032	737	251	229
Adagio (temps lié, arabesque, pas debourrée, balancé)	1169	588	213	167
Grand jeté (battement développé, chassé, grande jeté développé, arabesque, fouetté sauté, jeté en tournant)	2692	718	272	147

can be argued that the framework overestimates the available information of human motion in high-resolution sampling rate, as human motion is generally associated with low frequencies [34, 107].

To understand the effect of temporal accuracy and synchrony, the TPs were compared with and without CTW. In some sequences the TPs are substantially reduced. A notion of worth is related to the sequence which had the highest TP, in which the CTW did not have a large increasing effect. It can be argued that in full-body motions of the highest temporal accuracy, the computationally improved temporal accuracy by CTW does not over-emphasize the TPs. In other words, the temporal accuracy is already very high with raw data. To assess the impact of the loss of temporal information via CTW, it is important to compute the throughput with and without CTW when synchrony and timing are critical.

In addition, the individual limb contributions to the total TPs was estimated. The raw TP values (without dimension reduction) were averaged across all ballet dances. Estimation showed that the hands gave the largest contribution to the throughput, 16.3 % and 18.3 % for left and right hand respectively. Left leg contribution was much lower than right leg contribution,

Table 6.2: The throughputs of the ballet data sequences with 60 FPS. #dims denote the number of dimensions after the dimension reduction.

Sequence	TP		#dims	TP (Var-GP-LVM <sup>1</sup> )		#dims
	raw	PCA	PCA	CTW	No CTW	Var-GP-LVM
Tombé pas de bourrée, italian fouetté, piqué turn, jeté en tournant	2990	753	43	271	15	12
Adagio (passé devant développé, arabesque, écarté devant pas de bourré)	115	46	62	5.2	5.2	2
Petit jeté (glissade jeté, ballotté, ballon, entrechat, assemblé)	2086	419	42	103	8.1	5
Petit jeté (temps de cuisse, sissonne devant fermée, derrière fermée, sissonne ouvert pas de bourrée)	2141	489	42	1.4	3.1	5
Adagio (temps lié, arabesque, pas debourrée, balancé)	1657	546	58	157	12	12
Grand jeté (battement développé, chassé, grande jeté développé, arabesque, fouetté sauté, jeté en tournant)	1927	445	42	9.0	4.0	2

8.1 % vs. 12.1 %. Surprisingly, left leg had the lowest average TP features, and left thigh gave the lowest raw average TP value, 15.3 bps or 0.74 %. The highest raw average TP was from the right finger, 139.2 bps or 6.7 % of overall TP.

Features for the head, upper and lower body had much lower values: respectively 43 bps, 40 bps and 36 bps on average. The analysis shows a laterality effect in accordance with left or right hand preference, and probably also the most important body features in the aspect of professional ballet dance.

<sup>1</sup>Bayesian GP-LVM with ARD kernel dimension reduction [98].

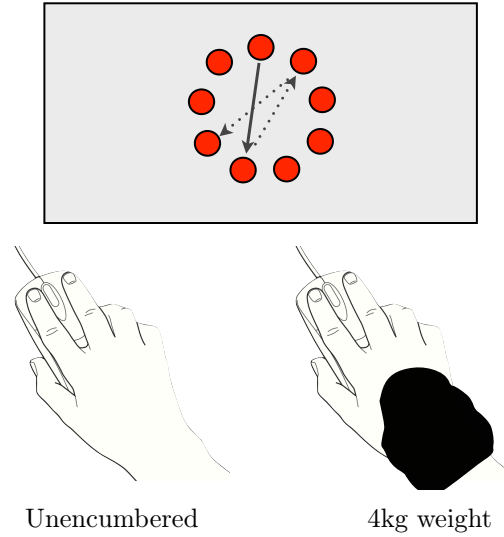


Figure 6.1: The setup of a cyclical tapping task in two conditions: with and without an additional 4 kg weight.

## 6.2 Cyclical Tapping With a Mouse

In order to compare our metric to the Fittsian metric, a well-studied variant of Fitts paradigm was replicated: a cyclical tapping experiment with a mouse, pictured in the Figure 6.1. The intent is to show that while the user does not intent to express information while moving the pointer to the target, the user still spends most of the total time on the way to each target. Thus, our metric complements the Fittsian metric by revealing the effect of variability in the trajectories.

The prediction was that TPs would be higher in cases where the repeated trajectories were close to each other. In the experiment’s paradigm, decreasing target width  $W$  should increase TP. Also, in contrast to the Fittsian metric, the shape of the curvature on turning towards next target should affect the TP as well, because drastic “spiky” turns should result in large residuals with the auto-regressive model.

Part of the experiment was also repeated with an added wrist weight of 4 kg. Also, the effect of a constant time task was investigated with a metronome. Without a metronome, the task was to be completed as quickly and accurately as possible. The subject (in this case, author) was provided plenty of practice time, also including gesturing with the wrist weight. The experiments started when the subject was confident in using the wrist weight, also with the additional use of a metronome. There was several trials with

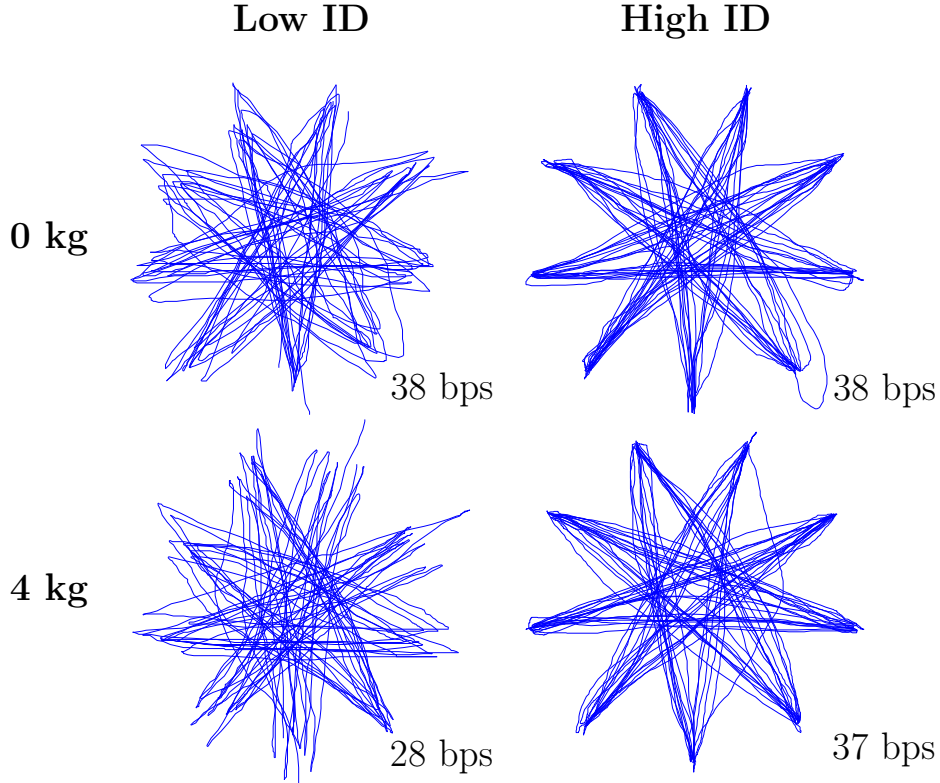


Figure 6.2: Movement trajectories (size normalized) with respective TP values in a cyclical tapping task with mouse.

every trial consisting of clicking through the cyclical circle targets three times.

First, the effect of weight was studied with random distance  $D$  and width  $W$  values, from a range of  $2.6 \leq ID \leq 6.1$ , according to Fitts-ID Equation (2.3). When effective width  $W_e$  was used as  $W$ , Fitts-TPs (2.8) for the unencumbered and 4 kg weight conditions were respectively 3.80 bps 2.45 bps (with Pearson correlation of  $R^2 = 0.90$  and  $R^2 = 0.93$ ). This agrees with the range of Fitts-TPs in previous studies [95]. As expected, the wrist weight had a decreasing effect to the TPs. The Fitts-TPs were calculated using only the separate traces from target to target, separated by the mouse clicks when turning during the cyclical clicking.

Second, the trajectories from conditions  $ID = 2.6$  and  $ID = 6.1$  were compared with unencumbered and encumbered states. For this experiment, the complete traces from each trial were used. The TPs were averages from pairwise comparisons, and no dimension reduction or preprocessing was needed in this case. The movement traces are shown in Figure 6.2. In unencumbered state, the TPs were almost the same, 38 bps and 38 respectively for low-ID



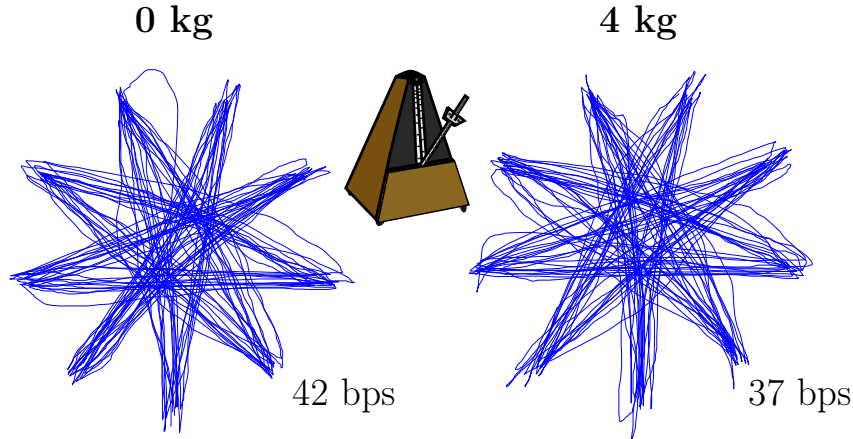


Figure 6.3: Movement trajectories and throughputs when a constant inter-click interval of 1.00 s was enforced ( $ID = 4.1$ ).

and high-ID conditions. When the wrist weight was added, it did not affect the TP in high-ID case (37 bps) but the TP in low-ID case fell to 28 bps.

It seems that the fast motion in the low-ID condition compensated the lower accuracy compared to high-ID case. It was observed that the subject rotated the hand in the high-ID case before starting the motion towards the next target, which compensated the slower movement speed, resulting in more closely aligned traces. This premovement aiming was not needed in the low-ID case where the targets were larger. Also, it can be argued that the added weight made the hand motion more steady, which resulted in smoother movement even though the average velocity was lower compared to low-ID case.

These experiments show that although the two different TP metrics can be used in the same experiment. However, a comparatively higher Fittsian TP (2.8) does not imply a higher TP from the new metric.

Surprisingly, as the weight did not decrease the TP in the high-ID condition, it was hypothesized that the subject compensated the added weight as there was no need to be hasty and error-prone in the aiming task. So, a follow-up experiment was made with a metronome, keeping the movement time at a constant 1.00 s. The ID condition was chosen to be  $ID = 4.1$ , which was still a manageable tapping task without too high error-rate even with the added weight. After proper practice, three trials were made in both unencumbered and encumbered states. The overlaid movement traces are shown in Figure 6.3.

It is evident that the increased tempo made the task significantly harder

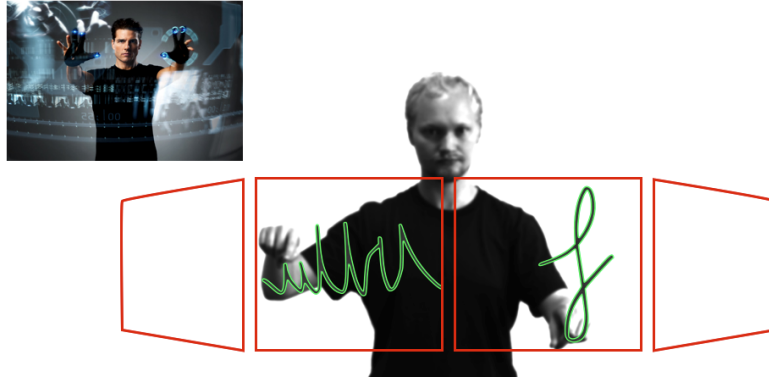


Figure 6.4: The bimanual in-air gesturing task, inspired by the film *Minority Report* (© 2002 DreamWorks).

with the added weight, which manifested in lower TP and more varying traces (average MT 1005 ms, target hitrate 89.5 %). Also, the tempo made the average TP in the unencumbered state quite high compared to the other conditions without a metronome (average MT 999.1 ms, target hitrate 95.0 %).

### 6.3 Bimanual In-Air Gesturing

To demonstrate the TP metric in the research of feasible interfaces, an experiment of in-air gesturing was carried. This kind of interface gesturing is now infamous because of the film *Minority Report* (2002) although HCI research has shown that bimanual continuous (asymmetric) control suffers from interference effects [14]. Inspired by the film, the subject (author) was instructed to gesture with both hands simultaneously, and the active interface space was divided into four parts in the field of view of  $120^\circ$ , see Figure 6.4. As the hand position in bimanual gesturing is critical, the experiment showed how the performance changed with the distance between the hands or if the gestures of the hands were switched.

In the experiment, the subject signed his name with one hand and simultaneously made another continuous gesture, which was asymmetric to the signature. As the signature is a typical *overlearned motor act*, it should retain some invariance when transferred from a familiar context to another [77]. Here, the signature gesture is transferred to an unfamiliar context and location of in-air space. Also, the signature gesture should act as a surrogate to the other learned gesture. Examples of the best and worst performances according to TP values are in Figure 6.5, where repetitions are superimposed

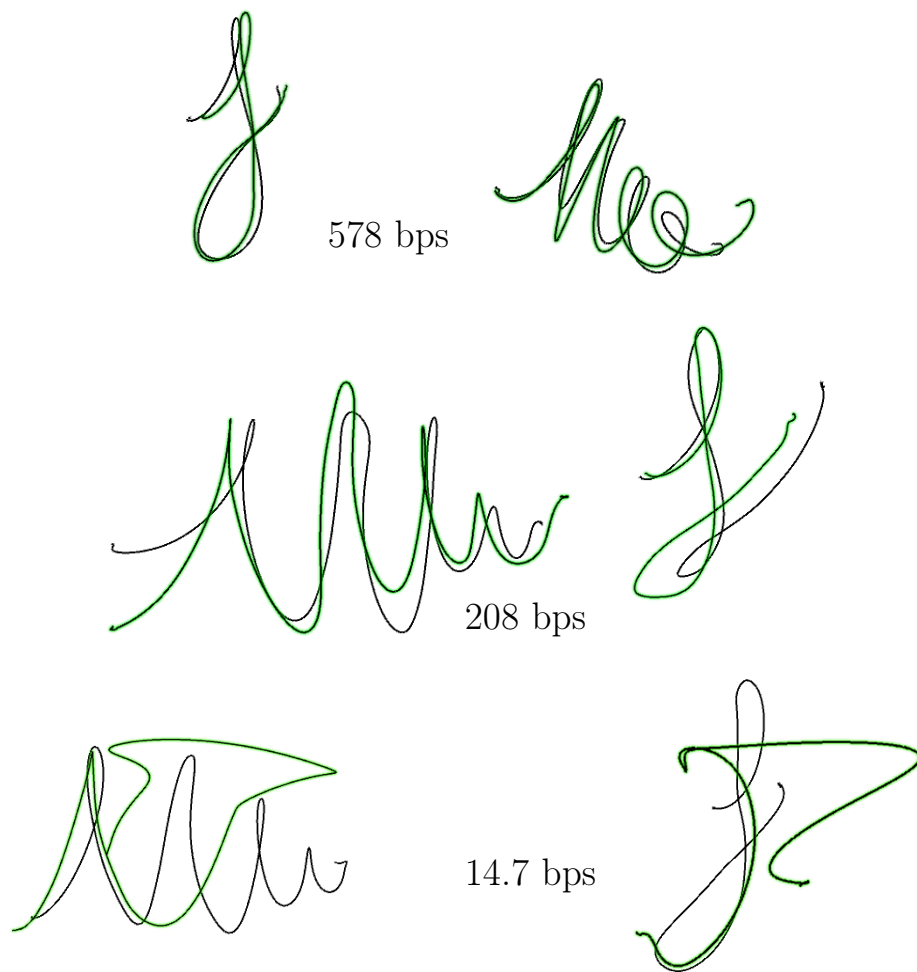


Figure 6.5: Examples of bimanual gesture performances with respective TP values. Repetitions are superimposed on top of each other. TOP: The performance with the highest TP. MIDDLE: A performance with a mediocre TP. BOTTOM: A failed performance with a very low TP. The gestures are not in scale.

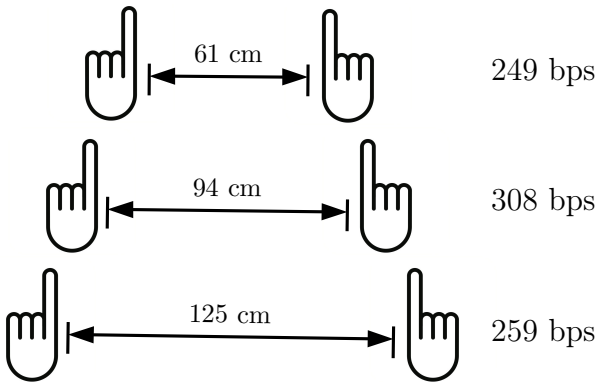


Figure 6.6: Average throughputs of bimanual control in different hand distances.

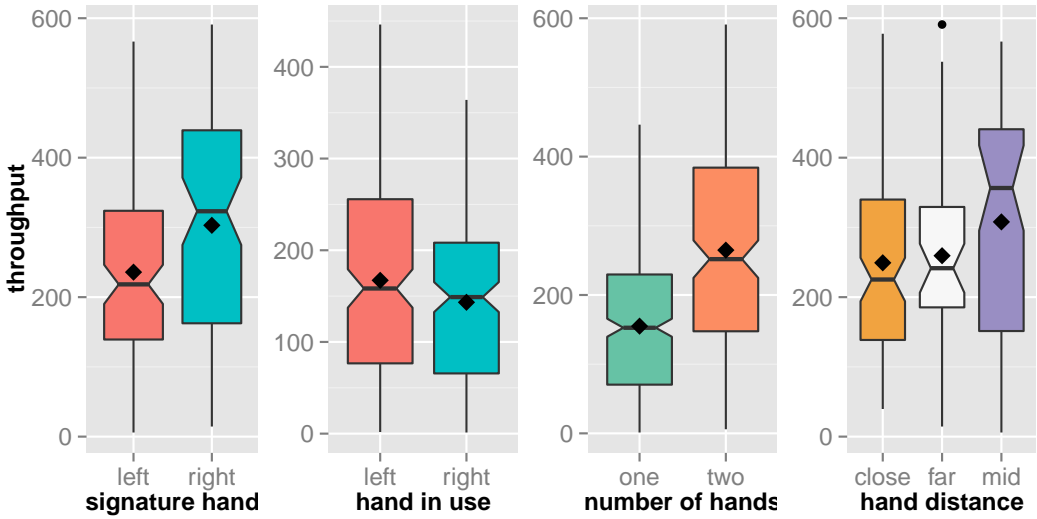


Figure 6.7: Boxplots of bimanual task throughputs in different conditions. The diamond points denote means.

Table 6.3: The p-values of population tests

Condition	p-value	
	MWW <sup>2</sup>	t-test <sup>3</sup>
Signature on left/right	0.0028	0.0025
Left/right hand	0.036	0.018
One/two hands	$< 2.2 \cdot 10^{-16}$	$< 2.2 \cdot 10^{-16}$

on top of each other.

Four observations were made in this study. First, throughputs (reduced with PCA) were slightly higher in the dominant hand, 167 bps vs. 143 bps. Second, the subject could express more information with both hands, the average throughput was 265 bps for both hands, while 155 bps for a single hand. Third, changing the other gesture to a totally new one hindered the performance and the throughputs decreased by about 100 bps on average. This possibly implies that without extensive learning, the subject's bimanual performance throughput is about the same level as in single-hand performance. Fourth, a quite surprising observation was that if hands were close together it also hindered the bimanual performance. This phenomenon is known as *perceptual bimanual distraction*; seeing both hands at the same time distracts their synchronous asymmetric control [64]. In this experiment, this close distance was 61 cm on average. When the hands are further apart, on average 94 cm, there was less distraction. If however the hands were very far apart (125 cm on average), barely in the field of view, average TP was again lower, perhaps due to biomechanical difficulties.

The average TPs on different hand distances are shown in Figure 6.6. Figure 6.7 represents the throughput distributions with boxplots in different conditions. The diamond points denote the means, and the waist bands correspond to the median, and the bottom and top of the boxes correspond to the first and third quartiles (the 25th and 75th percentiles). The “whiskers” extend from the top and bottom to the highest and lowest values that are within  $1.5 \times \text{IQR}$  of the range, where IQR is the inter-quartile-range, or distance between the first and third quartiles. Data beyond the end of the whiskers are outliers and plotted as points. The “notches” around the median extend  $1.58 \times \text{IQR}/\sqrt{n}$ , which gives a roughly 95 % interval for comparing medians. See McGill et al. [63] for more details.

<sup>2</sup>Mann-Whitney-Wilcoxon test or Wilcoxon rank sum test with continuity correction.

<sup>3</sup>Welch two sample t-test.

Assuming normal distributions, *Welch two sample t-tests* [100] were made for different observations. As the assumption might not be valid, *Mann-Whitney-Wilcoxon tests* (MWW) [60] were also applied. The results are shown in Table 6.3. Although boxplots in Figure 6.7 do not show statistical significance in laterality, or the better performance (as in throughput) of either left or right hand, the MWW and t-test show that at least on significance level of 0.05, the null hypothesis of identical distributions can be rejected. Note that the two-handed throughput was significantly higher when the signature was made with the right hand, although the subject was left-handed. This might be due to the more extensive training the subject spent rehearsing the asymmetric control in this condition, although the subject's preferred hand for the signature was left. The results seem reasonable, but no concrete claims can be notwithstanding made, as there was just a single subject.

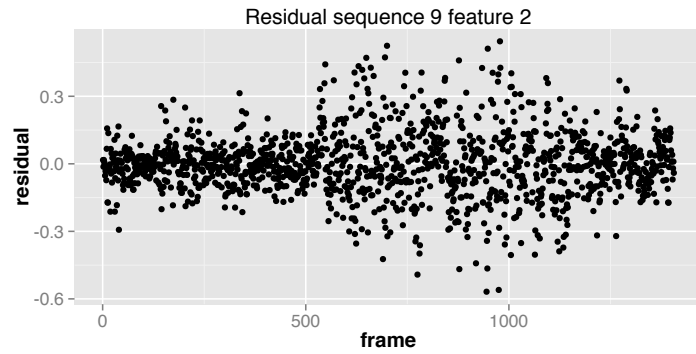
## 6.4 Metric's Assessment

### 6.4.1 Complexity Model Diagnostics

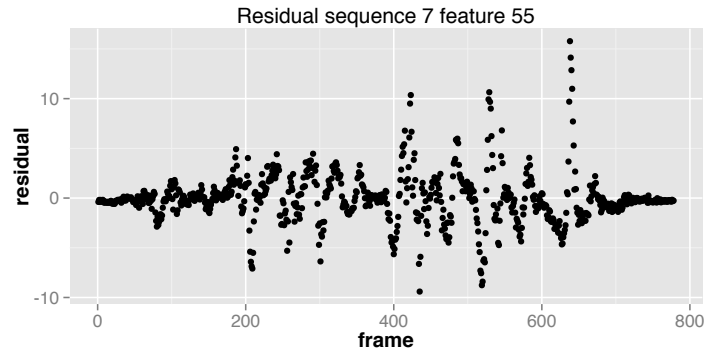
The second-order autoregressive model (AR(2)) chosen for the complexity estimation was very simple. As such, it has multiple features that can invalidate the assumptions and lead to non-satisfactory results. Some simple visual diagnostics can be used for checking model mis-specification, for example plotting the predicted model against the “true” motion sequence, or plotting the residuals against time. The Figure 6.8 contains plots of some residual vectors. Plot 6.8a shows a residual vector that does not contain any strong visible autocorrelation and can be considered to be similar to white noise. However, residual vectors similar to 6.8b are much more common with the motion data. This vector shows strong autocorrelation, which contains also many “spikes”, displaying that the model makes considerate overshooting in the prediction. Plot 6.8c also shows a more rare case of negative correlation of the residuals.

The mutual information estimation presented in Section 4.3.5 also presumes that the distributions of the residuals are normal. However, the histogram of the ballet data's residuals in Figure 6.9 shows that the distribution is visibly non-Gaussian and has very long tails. The distribution is more similar to Laplace distribution or Cauchy distribution than Gaussian distribution.

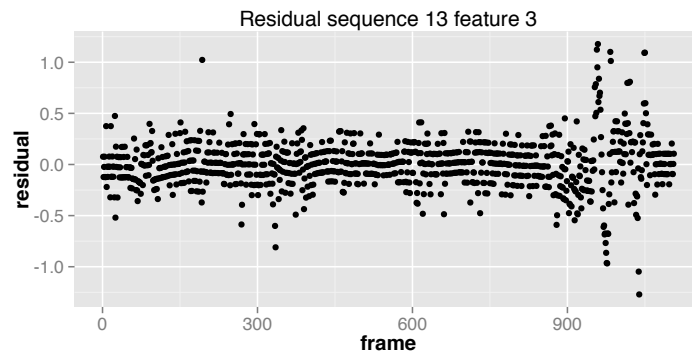
These visual diagnostics show that the AR(2) model does not perform very well, although it can be considered as adequate with the metric's eval-



(a) A residual vector showing no visible autocorrelation.



(b) A residual vector showing positive autocorrelation with spikes.



(c) A residual vector showing negative correlation.

Figure 6.8: Example plots of residuals of the autoregressive model. Plots (b) and (c) have dominating patterns.

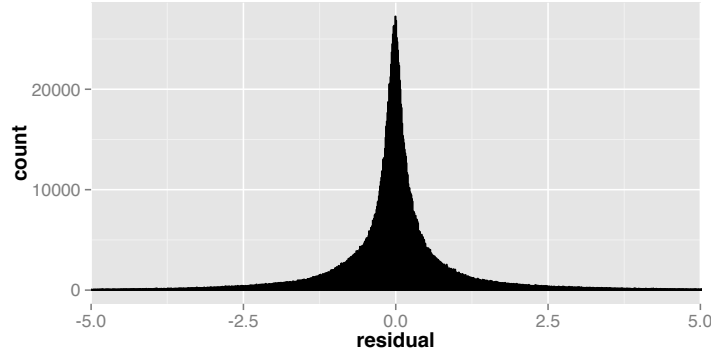


Figure 6.9: A histogram of ballet data's residuals with the autoregressive model. The distribution is visibly non-Gaussian with very long tails.

uation.

### 6.4.2 Frame Rate Manipulation

Regarding motion capture devices, high sample rates of 60 Hz and more should be enough to capture the accurate motion patterns, as the controlled human motion frequencies in (full-body) kinematics are typically 6–9 Hz [106, 107], though record class performers such as in tap dancing can reach tapping frequencies of 20 Hz or more temporarily [36]. The speed of motion can also cause inaccuracies with low sample rate. For example, in the ballet data the fastest movements are almost 9 m/s, and the delay of 60 Hz sample rate can lead to errors as large as 15 cm in the data. However, such drastic changes are only theoretical, and a 60 Hz sample rate should be enough to capture motion data for any practical purposes.

In theory, the throughput metric should be invariable to frame rate manipulation if there is no change to the amount of total mutual information in the sequences. However, the practical tests show that the current framework is not invariable with respect to framerate manipulation. An additional data set of 60 FPS was created by dropping half of the frames in the original 120 FPS Vicon ballet data. Using Butterworth filtering as described in Section 5.2, both data sets were smoothed to remove any remaining noise artifacts over 14 Hz. Comparison of the computed throughput values then revealed that on average, throughput was 40 % higher with 120 FPS data compared to 60 FPS data. The difference was even stronger after PCA dimension reduction; 120 FPS throughputs were 88 % higher than 60 FPS throughputs.

A second test was also carried out, this time interpolating and smoothing



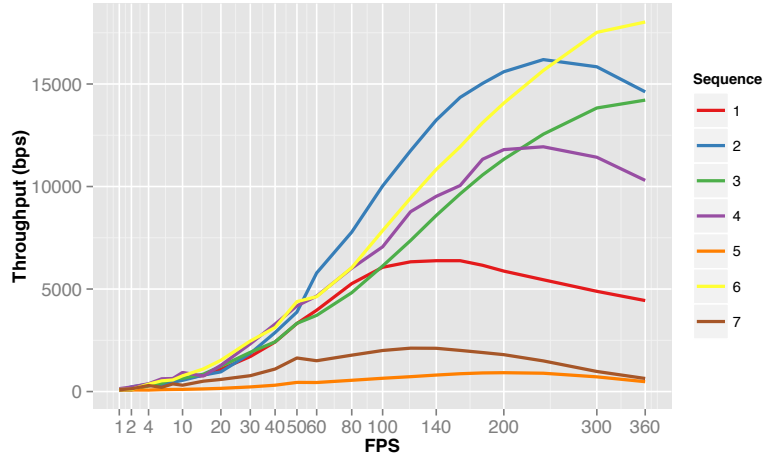


Figure 6.10: The throughputs of raw ballet data in different interpolated frame rates. The x-axis has a square root scale.

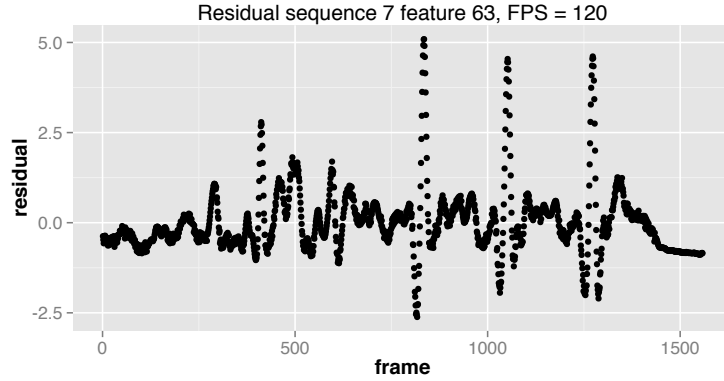
the original raw ballet data to multiple different frame rates. The throughputs of all sequences on different framerates are shown in Figure 6.10. The interpolation was done first with piecewise cubic splines and smoothing was performed after with Butterworth low-pass filter with cut-off frequency of 7 Hz.

The results in Figure 6.10 show that TP values grow until a certain FPS value, which is distinctive per sequence, after which the values will turn down. The different responses are difficult to explicate, but a residual analysis of the autoregressive model gives some pointers. For example, the sequences that have the peaks in higher framerates show more and larger residual “spikes” of overshooting. This points that the more drastic changes in the trajectories, the higher framerate is needed for the model to finally fit well to these changes.

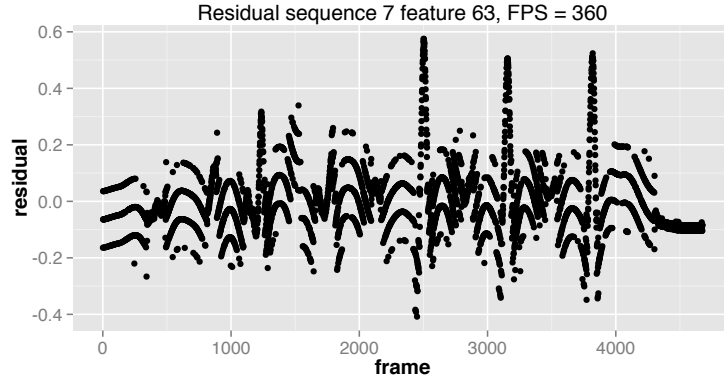
Also, as a general trend, oversampling the sequences creates lots of negative autocorrelation in the residuals, which displaces a better fit and smaller residuals. An example of this is shown in Figure 6.11. This seems to show that the second-order autoregressive model for complexity evaluation is not a very invariant or a robustly fitting model for the estimation of movement complexity.

### 6.4.3 Effect of Noise

When evaluating the throughput with respect to the quality of the data, the level of the instrument noise becomes important. Ideally, the TP metric would tolerate a level of noise that does not obscure controlled aspects of the



(a) A residual vector showing large “spikes” at 120 FPS.



(b) The same residual vector as above at 360 FPS.

Figure 6.11: An example of the effect of frame rate manipulation.

movement. However, especially MOCAP data can contain noise of varying levels, which depends on the methods of tracking and pose estimation. In the experiments, MOCAP data from *Vicon* system had negligible amount of noise, but for example *PhaseSpace* data had visible problems tracking all the markers and the output data contained inherent noise.

The sensitivity of the TP metric was tested by adding artificial Gaussian noise to the raw output data from *Vicon* system, comparing all different ballet motion sequences. Variance of the added noise was proportional to each data component’s variance. There was no dimension reduction, but one must note that especially the probabilistic models PPCA and GP-LVM would handle the noise factors better in the dimension reduction process.

The throughput was compared against a factor of Gaussian noise variance proportional to each component’s standard deviation, and the noisy data was aligned with CTW. The results are shown in Figure 6.12. The Figure shows

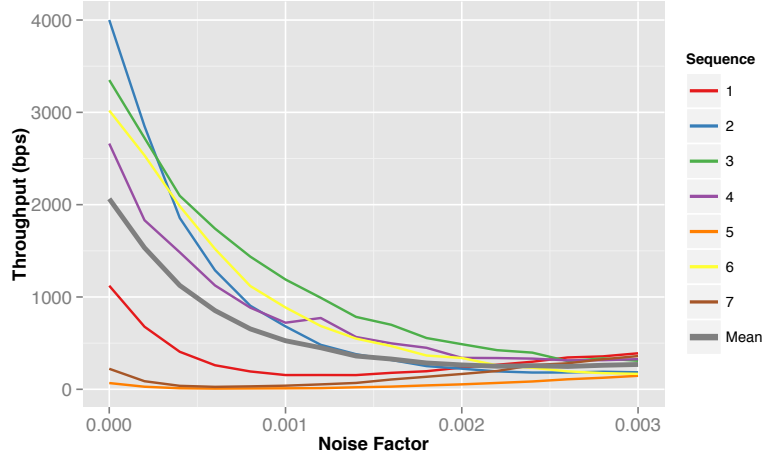


Figure 6.12: The effect of Gaussian noise added to raw ballet data. Noise with a factor of 1 equals the standard deviation in each feature.

that even a small amount of Gaussian noise can have a large decreasing effect on the throughput. On average, noise with a standard deviation of about  $5 \times 10^{-4}$  times the standard deviation of each component can halve the throughput, which corresponds to additional noise of submillimeter scale in the data. As the CTW step can also align strong noise artifacts, it is possible that TP values will occasionally increase due to temporal misalignment. This can be seen in Sequence 4 with noise factor 0.0012.

Because of the metric’s sensitivity to noise, a smoothing filtering is almost always required, even without visible artifacts. In the experiments shown here, cubic spline interpolation and Butterworth filtering have been used, which are commonly used in analysis of motion data [107].

#### 6.4.4 Static and Repetitive Motion

For a feasibility study, it was investigated whether simple motions produce low TPs, or how TP metric reacts to rapid monotonic motions. The following data with repetitions were collected via *PhaseSpace* system:

- Standing still
- Balancing one foot
- Rapid caging of the palms

Markers were placed on the full body, and in the examination of the hand, also on all fingers and both sides of the wrists. The throughputs were computed with CTW and basic GP-LVM with six latent dimensions.

As it was expected, balancing and standing still produced virtually no throughput, both TPs were lower than 0.25 bps. As a person is standing still, the residuals or the AR(2) model are negligible, that is, the model measures no complexity. In balancing on one foot, the swaying produces some complexity, but as the movements are poorly matched in repetitions, the produced TPs are negligible even with the use of CTW. This does not mean that balancing itself would not be motorically difficult, because it is [26]. However, for an external observer the balancing does not carry any additional information after the initial pose.

In contrast, the rapid caging of the palms yielded a very high average throughput of 288 bps, even with GP-LVM, although the motion is motorically trivial for an adult. This can be considered as a drawback of the AR(2) model because of its short-term memory. Human perception can easily detect repetitions which do not carry any additional information but the model evaluates the complexity of every instance of repetitions separately. However, as the temporal alignment is important in repetitions when measuring the correlations, the use of CTW makes a big difference. Without the CTW procedure, the average throughput was about 43 bps, so CTW increased the throughputs of this motion by a factor of 6.7.

## Chapter 7

# Discussion

This thesis, along with and extending the work by Oulasvirta et al. [74], has presented a novel metric for measuring the information capacity of full-body movements. The new metric has been inspired the work by Fitts [27] in the 1950s, and is thus an extended and generalized re-creation of Fitts’ law paradigm. It shares the intuitive premiss by Paul Fitts that the “*information capacity is limited only by the amount of statistical variability, or noise, that is characteristic of repeated efforts to produce the same response*”. However, the discrete restrictions of Fitts’ law are surpassed by considering

- the shape of a continuous movement trajectory as the source of information instead of the target width and distance, and
- the inaccuracy of the reproduced movement as the source of noise instead of end-point variation.

The Fittsian metric is basically a version of the minimum amount of *self-information* which an aiming task needs to be solved. As Fitts remarked, the index of difficulty specifies *the minimum information required on the average for controlling or organizing each movement*, and Fitts postulated an index of difficulty for simple amplitudal movement.

In contrast, the metric described in this thesis defines the *mutual information* of a reproduced task, which can be a more accurate metric in measuring the whole theoretical motor capacity, but can be grossly overestimated when the latent space of the motion is completely up to the subject herself with all available degrees of freedom, and thus the space is difficult to define. Nonetheless, we may assume that in a Fittsian aiming tasks, the “true” information capacity lies somewhere between the Fitts’s TP and ours.

The metric allows the examination of any scenario wherein users’ motion can be represented as time series features, from mouse movements to full-body motion. Any kind of movement data can be analyzed, with the only

requirement being the repeatability of the movement. In contrast, the known extensions of Fitts' law from discrete to continuous movements are predictive models of the movement time [2, 53], and do not carry an interpretation in information theory, nor can deal with multi-feature arbitrary trajectories in 3D space.

As the metric is based on estimation of *mutual information* in viewed movements, it should not be confused with the intrinsic difficulty of performing the movement nor with the neuro-motor system's information capacity. In reality, relatively simple motor-cognitive tasks can produce high throughput values, and some relatively complex feats, such as balancing, produce zero throughput. As such, the metric is best understood as the rate of information available to an external observer from the sensor space, defined through the complexity and reproducibility of captured movements.

Although, as the metric is a serious generalization of Fitts's TP, it lacks the predictive capability of the Fitts's paradigm. As the Fitts's TP is proportional to the index of difficulty, it is relatively robust to the changes in the task parameters,  $W$  and  $D$ , under the Fitts' law. With more complex and variable motor control tasks, interpolation and extrapolation cannot be expected, because free-form movement directions may invoke very different control patterns even with slight changes in trajectories.

The only assumption the metric makes about the data is a constant framerate. This makes it suitable for a wide range of uses, for example, researchers can do analysis on touchscreen and MOCAP data in a similar way. On the other hand, the absolute TP values in the data space can be extremely high compared to the familiar TP ranges in Fitts' law literature. However, the high range is expected because the metric may use high-frequency multi-feature sampling of continuous movements, and also because the model has no prior models of the performer or the environment that would restrict the degrees-of-freedom. It has been shown that the metric's absolute estimates can be greatly reduced by searching for the latent motion space using unsupervised machine learning, although robust automatic nonlinear dimension reduction is still problematic.

## 7.1 Applications

The metric has many foreseeable user cases, but in this thesis work, mainly the *analysis of motor control and performance* has been shown to be possible. In the Chapter 6 about evaluation, it was shown that the metric is sensitive to some well-known effects in motor control, such as laterality, encumbered movement, change in performance objective such as speed, and perceptual

distraction of bimanual control. As such, the metric is a good tool for sensitive and objective analysis. The metric can also be used to analyze the contributions of different limbs in users' full-body movement control, and to expose the performance-affecting factors. The timing and synchrony of motions can be explored by removing the temporal alignment (time warping) step.

Other possible user cases for the metric are related to user interface research and exploration. For example, different input methods or interface designs which share the same sensor space can be compared with respect to the available motor capacity. Using multiple complex reference gestures that are reproducible with each user interface design, researchers can compare either novel or alternative solutions. As most movement spaces are too large to be examined exhaustively, complex overlearned patterns such as signatures can be used to represent performance that users could attain with considerate practise.

Also, the metric could provide a useful tool for user interface exploration. Essential input mechanisms need to be distinguishable and controllable. This implies high variability and reliability of the gestures. If gesture movements have a variation which is not under the control of a user, the information capacity of the channel will be diminished as some movements will be indistinguishable.

The metric also incorporates recognisability with complexity, which is valuable in authentication. The security of free-form gesture *passwords* would be limited by the achievable throughput, when the reference gesture is saved to the device memory. The security and memorability of such password authentication system has been recently studied by Sherman et al. [87], showing that the metric could provide a sensible authentication threshold for free-form passwords.

To help users to apply the metric, there is a dedicated Web service at <http://infocapacity.hiit.fi/>. It enables the users to make computations with the metric without the need for installing any software. The interface provides an ease of use even without knowledge in programming, as the user can select the most important parameters through the Web client. The results are then sent to the user by email.

## 7.2 Future Work

The most important goal for future work is to combine the currently separate steps of complexity estimation, temporal alignment, and dimension reduction under a single robust model. Possible cues are Shared GP-LVM with manifold

relevance determination [22, 25] and Bayesian filtering incorporation, similar to GP-BayesFilters [46].

Unlike the Fittsian TP, the computation process is also very intensive and time-consuming with CTW and GP-LVM, and especially so if the motion sequences are long. Approximate Bayesian inference methods would improve the speed of computation, but GP-LVM still has fundamental problems in scalability, although many fast inference methods exist for GP regression and classification with expectation propagation (EP) [42]. One promising direction is introducing stochastic variation inference (SVI) to GP-LVM, but the implementation is tricky [38]. Another efficient option is using distributed variational inference through MapReduce algorithm, but full utilization requires the availability of hundreds of CPU cores with a suitable framework [23, 32].

The estimation of sequence complexity is currently very naive. Better results could be achieved with Bayesian estimation filters, such as a Kalman filter [43]. As the transformation related to motion dynamics and general noisy MOCAP data are linear, it is possible to estimate the variance of dynamics with maximum likelihood estimation and derive recursive *sensitivity equations* [37, 99].

Even further improvement could be achieved if model's state space is informed with a skeletal model of dimensions and movement ranges of bones and joints. The lower bound of information capacity could also be researched through biomechanical simulations, where the inputs are strictly controlled.

The metric itself needs more studies and evaluation in its potential. The preliminary results presented here need additional validation with more test subjects. Application in sport sciences include training of complex motor schemas with reference motions. Potential new diagnostic tools could be applied in clinical healthcare, for example, changes in the motor capacity could offer an early identification process of neurological disorders related to motor dysfunction, and monitoring the neuroplastic recovery after lesions and other impairments. Also, applications in user authentication have already been studied [87], but a robust, effective and comprehensive framework for mobile devices needs more work.



## Chapter 8

# Conclusions

This thesis work is an extended representation of the framework originally presented in [74]. The work is motivated by the limitations of Fitts' law [27], which has otherwise been very successful paradigm in human-computer interaction. With Fitts' law, it is possible to compare and evaluate the effectiveness of different input methods that are based on aiming movements. However, Fitts' law is restricted to discrete aiming tasks, where only the variation at the end-point matters. The experimental tasks are also predescribed and limited to one-limb movements. There exists extensions to Fitts' law that surpass some restrictions, however, they do not have a parallel interpretation in information theory which the original Fitts' law has.

The object of this work is to design a new metric, that has a footing in information theory, and generalizes the Fitts' law to free-form and full-body movement tasks. The development of motion capture technology, touch-screens, and depth image cameras has enabled the researchers to capture human motion more effectively, feasibly, and inexpensively in both 2D and 3D. In this thesis, different motion data acquisition methods have been used to collect data for reference, experimental evaluation, and demonstration purposes. The metric has been demonstrated to be usable with various sensor devices, and motion capture data provides the best format for extracting the information of human motion.

The new metric has been assessed that it responds fairly as it was supposed to. It does not compare straight-forwardly to the Fittsian throughput metric, but shares some similarities and extends the original Fitts's paradigm. Theoretically, the metric measures the throughput in bits per second, but as the calculation is done in the sensor space given by the input data, it is thus generally vastly over-estimated compared to the user's inherent motor capacity. The realistic throughput in joint user-sensor space needs careful model consideration and various dimension reduction tools from machine learning

field.

Achieving the goal of constructing a comprehensive and reliable measure of information capacity of human motion would lead to a wealth of useful knowledge about the human motor system and human-computer interaction. Concrete utility is to be seen, for instance, in the study of human-computer interaction with novel interface designs. Other possible fields of interest would be kinesiology, sports science, health diagnostics, and user authentication systems.

# Bibliography

- [1] Johnny Accot and Shumin Zhai. Beyond fitts' law: Models for trajectory-based hci tasks. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, CHI '97, pages 295–302, New York, NY, USA, 1997. ACM. ISBN 0-89791-802-9. doi: 10.1145/258549.258760. URL <http://doi.acm.org/10.1145/258549.258760>. (Citation on page 9.)
- [2] Johnny Accot and Shumin Zhai. Performance evaluation of input devices in trajectory-based tasks: An application of the steering law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 466–472, New York, NY, USA, 1999. ACM. ISBN 0-201-48559-1. doi: 10.1145/302979.303133. URL <http://doi.acm.org/10.1145/302979.303133>. (Citations on pages 9 and 58.)
- [3] Nabil Ali Ahmed and D.V. Gokhale. Entropy expressions and their estimators for multivariate distributions. *Information Theory, IEEE Transactions on*, 35(3):688–692, 1989. (Citation on page 29.)
- [4] Paul Annesley and Github Community. Pheanstalk, 2008. URL <https://github.com/pda/pheanstalk/>. Accessed 12.02.2014. (Citation on page 38.)
- [5] Paul Atkinson. The best laid plans of mice and men: the computer mouse in the history of computing. *Design issues*, 23(3):46–61, 2007. (Citation on page 4.)
- [6] Fred Attneave. Applications of information theory to psychology: A summary of basic concepts, methods, and results. 1959. (Citation on page 2.)
- [7] Richard Bellman. *Dynamic programming*. Dover Publications, Mineola, NY, USA, 2003. ISBN 0486428095. (Citation on page 18.)

- [8] Christopher M. Bishop and Nasser M. Nasrabadi. *Pattern recognition and machine learning*. Springer New York, 2006. (Citation on page 21.)
- [9] Joshua Blake and OpenKinect Community. OpenKinect libfreenect software, 2010. URL [http://openkinect.org/wiki/Main\\_Page](http://openkinect.org/wiki/Main_Page). Accessed 20.1.2014. (Citation on page 36.)
- [10] Ludwig Boltzmann. Ueber die Natur der Gasmoleküle. *Annalen Der Physik*, 236:175–176, 1877. doi: 10.1002/andp.18772360120. (Citation on page 28.)
- [11] George EP Box, Gwilym M Jenkins, and Gregory C Reinsel. *Time series analysis: forecasting and control*. Wiley, 4 edition, 2008. ISBN 978-0470272848. (Citation on page 27.)
- [12] David R Brillinger. Some data analyses using mutual information. *Brazilian Journal of Probability and Statistics*, 18(6):163–183, 2004. (Citation on page 33.)
- [13] Joaquin Quiñonero Candela and Carl Edward Rasmussen. A Unifying View of Sparse Approximate Gaussian Process Regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005. (Citation on page 26.)
- [14] Stuart K. Card, William K. English, and Betty J. Burr. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a crt. *Ergonomics*, 21(8):601–613, 1978. (Citations on pages 4, 8, and 16.)
- [15] Reid Chesterfield, Andrew Johnson, Charlie Lang, Matthew Stainer, and Jonathan Ziebarth. Solution-coating technology for amoled displays. *Information Display Magazine*, January 2011. (Citation on page 16.)
- [16] Rudolf Clausius. Über verschiedene für die anwendung bequeme formen der hauptgleichungen der mechanischen wärmetheorie. *Annalen der Physik*, 201(7):353–400, 1865. (Citation on page 28.)
- [17] CMU Graphics Lab. Motion Capture Database. URL <http://mocap.cs.cmu.edu/>. Accessed 20.1.2014. (Citation on page 12.)
- [18] Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012. (Citations on pages 29, 75, 77, and 78.)

- [19] Credo Interactive Inc. MeGa MoCap V2, 2004. URL <http://www.charactermotion.com/products/powermoves/megamocap/>. Accessed 20.1.2014. (Citation on page 12.)
- [20] Lehel Csató. *Gaussian processes: iterative sparse approximations*. PhD thesis, Aston University, 2002. (Citation on page 26.)
- [21] Lehel Csató and Manfred Opper. Sparse On-Line Gaussian Processes. *Neural Computation*, 14:641–668, 2002. doi: 10.1162/089976602317250933. (Citation on page 26.)
- [22] Andreas Damianou, Carl Ek, Michalis Titsias, and Neil Lawrence. Manifold relevance determination. *arXiv preprint arXiv:1206.4610*, 2012. (Citation on page 60.)
- [23] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: Simplified data processing on large clusters. *Commun. ACM*, 51(1):107–113, January 2008. ISSN 0001-0782. doi: 10.1145/1327452.1327492. URL <http://doi.acm.org/10.1145/1327452.1327492>. (Citation on page 60.)
- [24] Heiko Drewes. Only one fitts’ law formula please! In *CHI ’10 Extended Abstracts on Human Factors in Computing Systems*, CHI EA ’10, pages 2813–2822, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-930-5. doi: 10.1145/1753846.1753867. URL <http://doi.acm.org/10.1145/1753846.1753867>. (Citation on page 9.)
- [25] Carl Henrik Ek and Philip HS Torr Neil D Lawrence. *Shared Gaussian Process Latent Variable Models*. PhD thesis, PhD thesis, 2009. (Citation on page 60.)
- [26] Roger M. Enoka. *Neuromechanics of human movement*. Human kinetics, 2008. (Citation on page 56.)
- [27] Paul M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954. (Citations on pages 2, 4, 5, 8, 9, 57, 61, and 77.)
- [28] Paul M. Fitts and James R. Peterson. Information capacity of discrete motor responses. *Journal of experimental psychology*, 67(2):103, 1964. (Citation on page 4.)
- [29] Paul M. Fitts and Barbara K. Radford. Information capacity of discrete motor responses under different cognitive sets. *Journal of Experimental Psychology*, 71(4):475, 1966. (Citation on page 7.)

- [30] Barak Freedman, Alexander Shpunt, Meir Machline, and Yoel Arieli. Depth mapping using projected patterns, April 2 2008. US Patent App. 12/522,171. (Citation on page 14.)
- [31] Ben Fry, Casey Reas, and Processing Community. Processing programming language, 2001. URL <http://processing.org/>. Accessed 20.1.2014. (Citation on page 36.)
- [32] Yarin Gal, Mark van der Wilk, and Carl E Rasmussen. Distributed variational inference in sparse gaussian process regression and latent variable models. *arXiv preprint arXiv:1402.1389*, 2014. (Citation on page 60.)
- [33] Varun Ganapathi, Christian Plagemann, Daphne Koller, and Sebastian Thrun. Real time motion capture using a single time-of-flight camera. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 755–762. IEEE, 2010. (Citation on page 14.)
- [34] ACRMDOG Godfrey, R Conway, D Meagher, and G ÓLaighin. Direct measurement of human movement by accelerometry. *Medical engineering & physics*, 30(10):1364–1386, 2008. (Citation on page 41.)
- [35] Daniel Grest, Jan Woetzel, and Reinhard Koch. Nonlinear body pose estimation from depth images. In *Pattern Recognition*, pages 285–292. Springer, 2005. (Citation on page 14.)
- [36] Guinness World Records and Anthony Morigerato. Tap dancing – most taps in a minute, 2011. URL <http://www.guinnessworldrecords.com/records-3000/tap-dancing-most-taps-in-a-minute/>. Accessed 15.5.2014. (Citation on page 52.)
- [37] Narendra K Gupta and Raman K Mehra. Computational aspects of maximum likelihood estimation and reduction in sensitivity function calculations. *Automatic Control, IEEE Transactions on*, 19(6):774–783, 1974. (Citation on page 60.)
- [38] James Hensman, Nicoló Fusi, and Neil D. Lawrence. Gaussian processes for big data. *CoRR*, abs/1309.6835, 2013. (Citation on page 60.)
- [39] Errol R. Hoffmann. Which Version/Variation of Fitts’ Law? A Critique of Information-Theory Models. *Journal of Motor Behavior*, 45(3):205–215, 2013. doi: 10.1080/00222895.2013.778815. URL <http://www.tandfonline.com/doi/abs/10.1080/00222895.2013.778815>. PMID: 23581725. (Citation on page 9.)

- [40] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933. (Citation on page 21.)
- [41] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936. (Citation on page 17.)
- [42] Pasi Jylänki et al. *Approximate Bayesian Inference Methods for Regression and Classification with Gaussian Processes and Neural Networks*. PhD thesis, Aalto University, 2013. (Citation on page 60.)
- [43] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45, 1960. (Citation on page 60.)
- [44] Kouros Khoshelham. Accuracy analysis of kinect depth data. In *ISPRS workshop laser scanning*, volume 38, page 1, 2011. (Citations on pages 14 and 15.)
- [45] Steffen Knoop, Stefan Vacek, and Rüdiger Dillmann. Sensor fusion for 3d human body tracking with an articulated 3d body model. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 1686–1691. IEEE, 2006. (Citation on page 14.)
- [46] Jonathan Ko and Dieter Fox. Learning GP-BayesFilters via Gaussian process latent variable models. *Autonomous Robots*, 30(1):3–23, 2011. (Citation on page 60.)
- [47] Nils Krahnstoever and Rajeev Sharma. Articulated models from video. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–894. IEEE, 2004. (Citation on page 12.)
- [48] Nils Krahnstöver, Mohammed Yeasin, and Rajeev Sharma. Automatic acquisition and initialization of articulated models. *Machine Vision and Applications*, 14(4):218–228, 2003. (Citation on page 12.)
- [49] Neil Lawrence. GPmat Toolbox, 2013. URL <http://ml.sheffield.ac.uk/people/N.Lawrence/software.html>. Accessed 4.3.2014. (Citation on page 37.)
- [50] Neil D. Lawrence. Gaussian process models for visualisation of high dimensional data. In *Neural Information Processing Systems*, 2004. (Citation on page 25.)

- [51] Neil D. Lawrence. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *The Journal of Machine Learning Research*, 6:1783–1816, 2005. (Citations on pages 23 and 25.)
- [52] Leap Motion Inc. Leap Motion, 2013. URL <http://leapmotion.com/>. (Citation on page 15.)
- [53] Lei Liu and Robert van Liere. The effect of varying path properties in path steering tasks. In *Proceedings of the 16th Eurographics Conference on Virtual Environments & Second Joint Virtual Reality, EGVE - JVRC'10*, pages 9–16, Aire-la-Ville, Switzerland, Switzerland, 2010. Eurographics Association. ISBN 978-3-905674-30-9. doi: 10.2312/EGVE/JVRC10/009-016. URL <http://dx.doi.org/10.2312/EGVE/JVRC10/009-016>. (Citations on pages 9 and 58.)
- [54] David J. C. MacKay et al. Bayesian nonlinear modeling for the prediction competition. *Ashrae Transactions*, 100(2):1053–1062, 1994. (Citation on page 26.)
- [55] I. Scott MacKenzie. A note on the information-theoretic basis for fitts' law. *Journal of Motor Behavior*, 21(3):323–330, 1989. doi: 10.1080/00222895.1989.10735486. URL <http://www.tandfonline.com/doi/abs/10.1080/00222895.1989.10735486>. PMID: 15136269. (Citation on page 8.)
- [56] I. Scott MacKenzie. Fitts' law as a research and design tool in human-computer interaction. *Hum.-Comput. Interact.*, 7(1):91–139, March 1992. ISSN 0737-0024. doi: 10.1207/s15327051hci0701\_3. URL [http://dx.doi.org/10.1207/s15327051hci0701\\_3](http://dx.doi.org/10.1207/s15327051hci0701_3). (Citations on pages 4 and 8.)
- [57] I. Scott MacKenzie and Poika Isokoski. Fitts' throughput and the speed-accuracy tradeoff. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1633–1636, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-011-1. doi: 10.1145/1357054.1357308. URL <http://doi.acm.org/10.1145/1357054.1357308>. (Citations on pages 6 and 7.)
- [58] I. Scott MacKenzie, Abigail Sellen, and William A. S. Buxton. A comparison of input devices in element pointing and dragging tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, pages 161–166, New York, NY, USA, 1991.



- ACM. ISBN 0-89791-383-3. doi: 10.1145/108844.108868. URL <http://doi.acm.org/10.1145/108844.108868>. (Citation on page 8.)
- [59] Ian Scott MacKenzie. A Note on the Validity of the Shannon Formulation for Fitts' Index of Difficulty. *Open Journal of Applied Sciences*, 3:360, 2013. (Citation on page 9.)
- [60] H. B. Mann and D. R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, 18(1):50–60, 03 1947. doi: 10.1214/aoms/1177730491. URL <http://dx.doi.org/10.1214/aoms/1177730491>. (Citation on page 50.)
- [61] R. G. Marteniuk, C. L. MacKenzie, and D. M. Baba. Bimanual movement control: Information processing and interaction effects. *The Quarterly Journal of Experimental Psychology Section A*, 36(2): 335–365, 1984. doi: 10.1080/14640748408402163. URL <http://www.tandfonline.com/doi/abs/10.1080/14640748408402163>. (Citation on page 9.)
- [62] Chris McDonough and Mike Naberezny. Supervisor: A process control system, 2004. URL <http://supervisord.org/>. Accessed 9.10.2013. (Citation on page 38.)
- [63] Robert McGill, John W. Tukey, and Wayne A. Larsen. Variations of box plots. *The American Statistician*, 32(1):12–16, 1978. doi: 10.1080/00031305.1978.10479236. URL <http://amstat.tandfonline.com/doi/abs/10.1080/00031305.1978.10479236>. (Citation on page 49.)
- [64] Franz Mechsner, Dirk Kerzel, Guënthèr Knoblich, and Wolfgang Prinz. Perceptual basis of bimanual coordination. *Nature*, 414(6859):69–73, 2001. (Citation on page 49.)
- [65] Microsoft. Kinect, 2010. URL <http://www.xbox.com/kinect>. Accessed 10.1.2014. (Citations on pages 11 and 14.)
- [66] Microsoft. Kinect for windows sdk, 2013. URL <http://www.microsoft.com/en-us/kinectforwindows/>. Accessed 20.1.2014. (Citation on page 36.)
- [67] Thomas B Moeslund and Erik Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, 2001. (Citation on page 11.)

- [68] Thomas B Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2):90–126, 2006. (Citations on pages 11 and 12.)
- [69] Olaf Munkelt, Christof Ridder, David Hansel, and Walter Hafner. A model driven 3d image interpretation system applied to person detection in video images. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, volume 1, pages 70–73. IEEE, 1998. (Citation on page 12.)
- [70] Radford M. Neal. *Bayesian learning for neural networks*. PhD thesis, University of Toronto, 1995. (Citation on page 26.)
- [71] Oculus VR. Oculus Rift, 2012. URL <http://www.oculusvr.com/>. Accessed 10.1.2014. (Citation on page 11.)
- [72] OpenNI. OpenNI Framework, 2010. URL <http://www.openni.org/>. Accessed 20.1.2014. (Citation on page 36.)
- [73] OptoFidelity. OptoFidelity TPPT tester proved significant fails in Apple iPhone touch accuracy, 2013. URL <http://www.optofidelity.com/archives/3937>. Accessed 28.2.2014. (Citation on page 16.)
- [74] Antti Oulasvirta, Teemu Roos, Arttu Modig, and Laura Leppänen. Information capacity of full-body movements. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 1289–1298, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2466169. URL <http://doi.acm.org/10.1145/2470654.2466169>. (Citations on pages 2, 57, and 61.)
- [75] Anthony O'Hagan. Some bayesian numerical analysis. *Bayesian statistics*, 4:345–363, 1992. (Citation on page 23.)
- [76] Karl Pearson. LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901. (Citation on page 21.)
- [77] R.W. Pew. Performance of human operators in a three-state relay control system with velocity-augmented displays. *Human Factors in Electronics, IEEE Transactions on*, HFE-7(2):77–83, June 1966. ISSN 0096-249X. doi: 10.1109/THFE.1966.232327. (Citation on page 46.)

- [78] Ronald Poppe. Vision-based human motion analysis: An overview. *Computer vision and image understanding*, 108(1):4–18, 2007. (Citation on page 14.)
- [79] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition*. Prentice Hall, 1993. (Citation on page 18.)
- [80] Keith Rarick. Beanstalk, 2007. URL <http://kr.github.io/beanstalk/>. Accessed 9.10.2013. (Citation on page 38.)
- [81] Carl Edward Rasmussen and C.K.I. Williams. *Gaussian processes for machine learning. 2006*. The MIT Press, Cambridge, MA, USA, 2006. ISBN 0-262-18253-X. (Citation on page 23.)
- [82] J. Rasmussen. Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-13(3):257–266, May 1983. ISSN 0018-9472. doi: 10.1109/TSMC.1983.6313160. (Citation on page 2.)
- [83] Sam Roweis. EM algorithms for PCA and SPCA. *Advances in neural information processing systems*, pages 626–632, 1998. (Citation on page 21.)
- [84] Matthias Seeger, Christopher K. I. Williams, and Neil D. Lawrence. Fast forward selection to speed up sparse gaussian process regression. In *Workshop on AI and Statistics*, volume 9, page 2003, 2003. (Citation on page 26.)
- [85] Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948. (Citations on pages 8, 28, 76, and 77.)
- [86] Claude E Shannon. The bandwagon. *IRE Transactions on Information Theory*, 2(1):3, 1956. (Citation on page 2.)
- [87] Michael Sherman, Gradeigh Clark, Yulong Yang, Shridatt Sugrim, Arttu Modig, Janne Lindqvist, Antti Oulasvirta, and Teemu Roos. User-generated free-form gestures for authentication: Security and memorability. *arXiv preprint arXiv:1401.0561*, 2014. (Citations on pages 59 and 60.)
- [88] Pete Shinnars and Pygame Community. Pygame, 2000. URL <http://pygame.org/>. Accessed 16.10.2013. (Citation on page 36.)

- [89] Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013. (Citation on page 14.)
- [90] Matheen Siddiqui and Gérard Medioni. Human pose estimation from a single view point, real-time range sensor. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 1–8. IEEE, 2010. (Citation on page 14.)
- [91] Marius-Călin Silaghi, Ralf Plänkers, Ronan Boulic, Pascal Fua, and Daniel Thalmann. Local and global skeleton fitting techniques for optical motion capture. In *Modelling and Motion Capture Techniques for Virtual Environments*, pages 26–40. Springer, 1998. (Citation on page 12.)
- [92] E. Snelson and Z. Ghahramani. Sparse Gaussian Process Using Pseudo-inputs. In *Neural Information Processing Systems*, 2006. (Citation on page 26.)
- [93] Yang Song, Luis Goncalves, and Pietro Perona. Unsupervised learning of human motion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(7):814–827, 2003. (Citation on page 12.)
- [94] Sony. PlayStation Eye, 2007. URL <http://us.playstation.com/ps3/accessories/playstation-eye-camera-ps3.html>. Accessed 10.1.2014. (Citation on page 11.)
- [95] R. William Soukoreff and I. Scott MacKenzie. Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts’ law research in HCI. *International Journal of Human-Computer Studies*, 61(6):751 – 789, 2004. ISSN 1071-5819. doi: <http://dx.doi.org/10.1016/j.ijhcs.2004.09.001>. URL <http://www.sciencedirect.com/science/article/pii/S1071581904001016>. Fitts’ law 50 years later: applications and contributions from human-computer interaction. (Citations on pages 4, 8, and 44.)
- [96] Michael E. Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999. (Citations on pages 21 and 23.)

- [97] Michalis K. Titsias. Variational learning of inducing variables in sparse gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, volume 5, pages 567–574, 2009. (Citation on page 26.)
- [98] Michalis K. Titsias and Neil D. Lawrence. Bayesian Gaussian Process Latent Variable Model. *Journal of Machine Learning Research*, 9:844–851, 2010. (Citations on pages 26 and 42.)
- [99] Ville Juhana Väänänen et al. Gaussian filtering and smoothing based parameter estimation in nonlinear models for sequential data. Master’s thesis, Aalto University, 2012. (Citation on page 60.)
- [100] Bernard L Welch. The generalization of Student’s’ problem when several different population variances are involved. *Biometrika*, pages 28–35, 1947. (Citation on page 50.)
- [101] A.T. Welford. *Fundamentals of skill*. Methuen, New York, NY, USA, 1968. (Citation on page 6.)
- [102] Chris K.I. Williams. Gaussian processes for machine learning. Seminar Lecture, August 2007. URL <http://www.newton.ac.uk/webseminars/pg+ws/2007/bnr/bnrw01/0809/williams/>. Bayesian Nonparametric Regression: Theory, Methods and Applications. Accessed 4.3.2014. (Citation on page 24.)
- [103] Christopher K.I. Williams. Prediction with gaussian processes: From linear regression to linear prediction and beyond. In *Learning in graphical models*, pages 599–621. Springer, 1998. (Citation on page 23.)
- [104] Christopher K.I. Williams and Carl Edward Rasmussen. Gaussian processes for regression. 1996. (Citation on page 23.)
- [105] John Williamson and Roderick Murray-Smith. Rewarding the original: Explorations in joint user-sensor motion spaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’12, pages 1717–1726, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2208301. URL <http://doi.acm.org/10.1145/2207676.2208301>. (Citation on page 9.)
- [106] David A. Winter. *Biomechanics and motor control of human movement*. Wiley, 2009. (Citations on pages 2, 36, and 52.)

- [107] David A Winter, H Grant Sidwall, and Douglas A Hobson. Measurement and reduction of noise in kinematics of locomotion. *Journal of biomechanics*, 7(2):157–159, 1974. (Citations on pages 36, 41, 52, and 55.)
- [108] Robert Sessions Woodworth. Accuracy of voluntary movement. *The Psychological Review: Monograph Supplements*, 3(3):i, 1899. (Citation on page 6.)
- [109] Shumin Zhai. On the validity of throughput as a characteristic of computer input. IBM Research Report, 2002. (Citation on page 4.)
- [110] Shumin Zhai. Characterizing computer input with Fitts’ law parameters—the information and non-information aspects of pointing. *International Journal of Human-Computer Studies*, 61(6):791 – 809, 2004. ISSN 1071-5819. doi: <http://dx.doi.org/10.1016/j.ijhcs.2004.09.006>. URL <http://www.sciencedirect.com/science/article/pii/S1071581904001041>. Fitts’ law 50 years later: applications and contributions from human-computer interaction. (Citations on pages 4 and 6.)
- [111] Feng Zhou. Software for Canonical Time Warping, 2009. URL [http://www.f-zhou.com/ta\\_code.html](http://www.f-zhou.com/ta_code.html). Accessed 28.2.2014. (Citation on page 37.)
- [112] Feng Zhou and Fernando De la Torre. Canonical time warping for alignment of human behavior. In *Advances in Neural Information Processing Systems Conference (NIPS)*, pages 2286–2294, December 2009. (Citations on pages 18 and 19.)
- [113] Youding Zhu and Kikuo Fujimura. Constrained optimization for human pose estimation from depth sequences. In *Computer Vision-ACCV 2007*, pages 408–418. Springer, 2007. (Citation on page 14.)

## Chapter 9

# Appendix: Gaussian Channel Capacity

### 9.1 Channel Capacity

The information channel capacity of a discrete *memoryless* channel is

$$C = \max_{p(x)} I(X; Y) = \max_{p(x)} [H(Y) - H(Y|X)]. \quad (9.1)$$

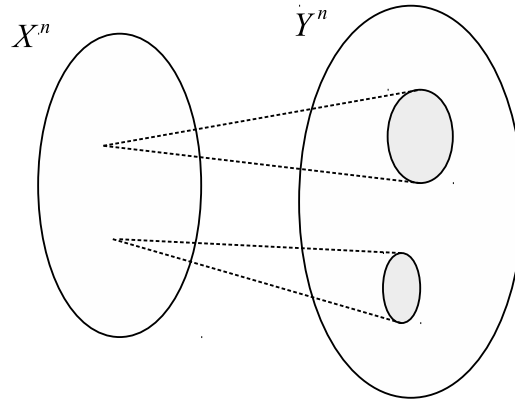


Figure 9.1: Information channels after transmitting a message with  $n$  symbols. The input space is  $X^n$  and the output space is  $Y^n$ . Adapted from [18].

The operational meaning to the definition of a capacity  $C$  is the number of bits one can transfer reliably over a channel. Because of innate channel noise, for each input  $n$ -sequence there are approximately  $2^{nH(Y|X)}$  possible  $Y$

sequences which are all approximately equally likely [85]. This is represented in Figure 9.1. For communication purposes, we wish to ensure that no two  $X$  sequences produce the same  $Y$  output sequence over the channel. Otherwise we will not be able to decide which sequence  $X$  was sent.

The total number of possible  $Y$  sequences is approximately  $2^{nH(Y)}$ , represented by the right big ellipse in Figure 9.1. This set has to be divided into sets of size  $2^{nH(Y|X)}$  which correspond to the different input  $X$  sequences. Then the total number of disjoint sets is less than or equal to

$$\frac{2^{nH(Y)}}{2^{nH(Y|X)}} = 2^{n(H(Y)-H(Y|X))} = 2^{nI(X;Y)}.$$

Thus we can send only approximately  $2^{nI(X;Y)}$  distinguishable sequences of length  $n$ . This outlines the upper bound of the channel capacity, but as Shannon's second theorem shows, any rate up to the bound (not necessarily inclusive) is achievable with an arbitrarily low probability of error. [18] [85]

## 9.2 Gaussian Channel

Let us suppose that information is sent over a channel that is subjected to additive white Gaussian noise. Then the output is

$$Y_i = X_i + Z_i, \quad (9.2)$$

where  $Y_i$  is the channel output,  $X_i$  is the channel input, and  $Z_i$  is zero-mean Gaussian noise with variance  $N$ . The noise  $Z_i \sim \mathcal{N}(0, N)$  is assumed to be independent of the signal  $X_i$  and it is independent and identically distributed (i.i.d.). Thus the output (and input) can take on a continuum of values.

Without further conditions, the channel capacity of a Gaussian channel may be infinite. If the noise variance is zero, the channel can transmit an arbitrary real number with no error. With no constraints on the input, we can choose an infinite subset of inputs arbitrarily far apart, and they are distinguishable at the output with arbitrarily small probability of error (see Figure 9.1).

A common assumption is that the average power of the channel is constrained. This means that for any codeword  $(x_1, x_2, \dots, x_n)$  transmitted over the channel, we require that

$$\frac{1}{n} \sum_{i=1}^n x_i^2 \leq P. \quad (9.3)$$



We can then define the information capacity of the channel as the maximum mutual information between input and noisy output over all distributions on the input that satisfy the power constraint. Expanding the  $I(X; Y)$ , we have

$$I(X; Y) = h(Y) - h(Y|X) \quad (9.4)$$

$$= h(Y) - h(X + Z|X) \quad (9.5)$$

$$= h(Y) - h(Z|X) \quad (9.6)$$

$$= h(Y) - h(Z), \quad (9.7)$$

since  $Z$  is independent of  $X$ . Because  $Z_i \sim \mathcal{N}(0, N)$ , the entropy of  $Z$  is  $h(Z) = \frac{1}{2} \log 2\pi e N$  according to the Equation (4.29). Also,

$$\mathbb{E}[Y^2] = \mathbb{E}[(X + Z)^2] = \mathbb{E}[X^2] + 2\mathbb{E}[X]\mathbb{E}[Z] + \mathbb{E}[Z^2] = P + N, \quad (9.8)$$

since  $X$  and  $Z$  are independent and  $\mathbb{E}[Z] = 0$ . Then  $h(Y)$  is bounded by  $\frac{1}{2} \log 2\pi e(P + N)$  because the normal distribution is the maximum-entropy distribution for a given variance. [18]

Applying this result to bound the mutual information, we get

$$I(X; Y) = h(Y) - h(Z) \quad (9.9)$$

$$\leq \frac{1}{2} \log 2\pi e(P + N) - \frac{1}{2} \log 2\pi e N \quad (9.10)$$

$$= \frac{1}{2} \log \left( 1 + \frac{P}{N} \right). \quad (9.11)$$

Hence, the information capacity of the Gaussian channel is

$$C = \max_{\mathbb{E}X^2 \leq P} I(X; Y) = \frac{1}{2} \log \left( 1 + \frac{P}{N} \right), \quad (9.12)$$

which is attained when  $X \sim \mathcal{N}(0, P)$  [85]. It can also be shown that this capacity is also the supremum of the rates achievable for the channel. [18]

Paul Fitts described his seminal work [27] as the *information capacity of the human motor system in controlling the amplitude of movement*. It was shown in Chapter 2 that Fitts's paradigm has an information theory interpretation with self-information.

The geometric plausibility of the aforementioned result can be shown as a sphere packing problem. For a codeword of length  $n$ , the received vector is normally distributed with mean equal to the true codeword and variance equal to the noise variance. With high probability, the received vector is contained in a sphere of radius  $\sqrt{n(N + \epsilon)}$  because the vector falls within

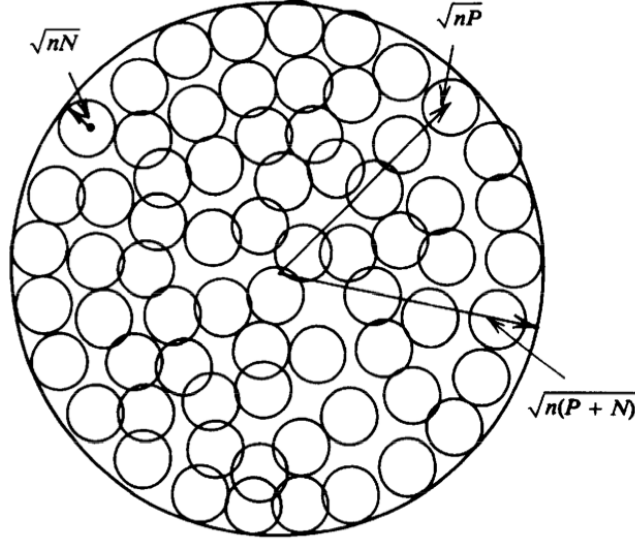


Figure 9.2: Sphere packing for the Gaussian channel. [18]

one standard deviation away from the mean in each direction, so the distance from the true codeword is the Euclidean distance  $\sqrt{\mathbb{E}[z_1^2 + z_2^2 + \cdots + z_n^2]} = \sqrt{nN}$ . If we assign everything within this sphere to the given codeword, we misdetect only if the output vector falls outside the sphere. Other codewords will have other spheres each with radius approximately  $\sqrt{n(N + \epsilon)}$ . The received vectors are limited in energy, so they all must lie in a sphere of radius  $\sqrt{n(P + N)}$ . The volume of an  $n$ -dimensional sphere is of form  $C_n r^n$ , where  $r$  is the radius of the sphere. The approximate maximum number of non-intersecting decoding spheres is therefore

$$\text{number of spheres} = \frac{C_n (n(P + N))^{\frac{n}{2}}}{C_n (n(N + \epsilon))^{\frac{n}{2}}} \approx 2^{\frac{n}{2}(1 + \frac{P}{N})}, \quad (9.13)$$

and the rate of code is thus  $\frac{1}{2} \log(1 + \frac{P}{N})$ . This idea is illustrated in Figure 9.2. This indicates that we cannot hope to send information at rates higher than  $C$  with low probability of error.